People + AI Research

Human-centered research and design to make AI partnerships productive, enjoyable, and fair

Building with sequence models

Responsibility and Prototyping with LLMs

LxLMS - 2025 Summer School

Lucas Dixon, co-lead of PAIR
ldixon@google.com
Google DeepMind

July 2025

PAIR | People + AI Research

Human-centered research and design to make AI partnerships productive, enjoyable, and fair

Boundary objects

"In sociology, a boundary object is information, such as specimens, field notes, and maps, **used in different ways by different communities.**"

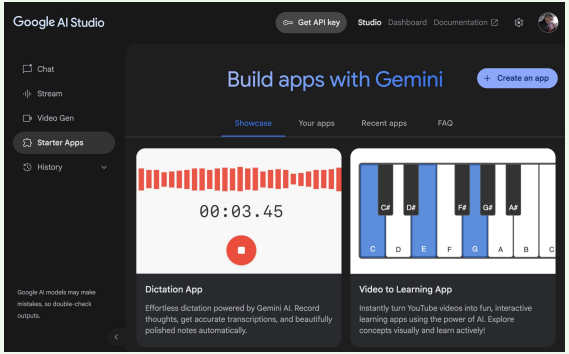
Star & Griesemer / as cited by Wikipedia

Example: architectural diagrams used to connect the designer to the builder to the future resident of a building.

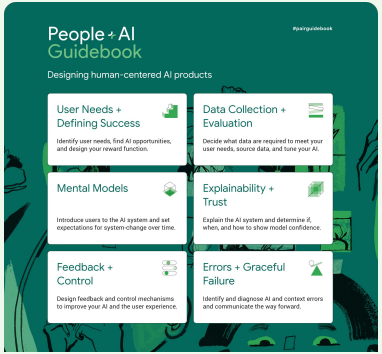
Open Source Tools & Platforms

to develop AI

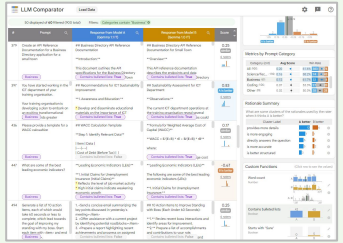
& develop with AI



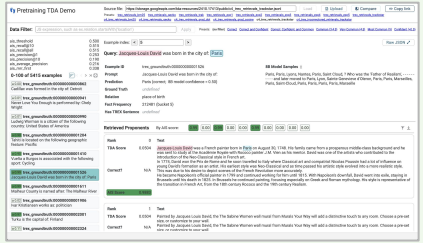
aistudio.google.com



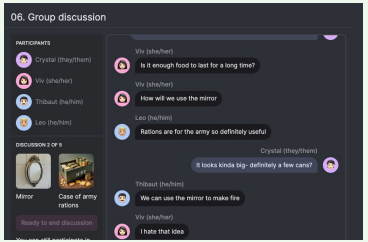
People+AI Guidebook



LLM Comparator
Compare LLMs



TrackStar
Training data attribution



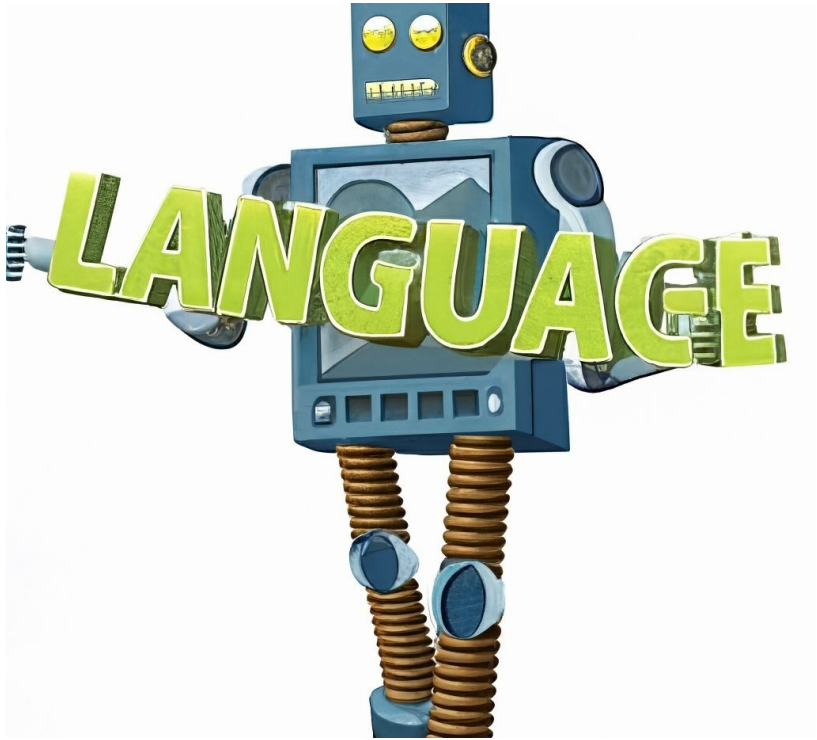
deliberate-lab
group + AI crowdsourcing

PAIR

More at pair.withgoogle.com/tools/

How to think about
Large Language Models?
(LLMs)

Actually, don't think about LLMs, think about LDMs:
(natural) **language driven models**
Language can control what AI does: APIs, make images, music, robots etc...



How I think about LLMs...

An interpreter (that can translate between languages, concepts, and styles)

LLMs

An improv comedian

A fuzzy database of the web

LLMs are VERY easy to customize

This is what makes them "general"

Sparks of Artificial General Intelligence: Early experiments with GPT-4

Sébastien Bubeck Varun Chandrasekaran Ronen Eldan Johannes Gehrke
Eric Horvitz Ece Kamar Peter Lee Yin Tat Lee Yuanzhi Li Scott Lundberg
Harsha Nori Hamid Palangi Marco Tulio Ribeiro Yi Zhang

Microsoft Research

Abstract

Artificial intelligence (AI) researchers have been developing and refining large language models (LLMs) that exhibit remarkable capabilities across a variety of domains and tasks, challenging our understanding of learning and cognition. The latest model developed by OpenAI, GPT-4 [Ope23], was trained using an unprecedented scale of compute and data. In this paper, we report on our investigation of an early version of GPT-4, when it was still in active development by OpenAI. We contend that (this early version of) GPT-4 is part of a new cohort of LLMs (along with ChatGPT and Google's PaLM for example) that exhibit more general intelligence than previous AI models. We discuss the rising capabilities and implications of these models. We demonstrate that, beyond its mastery of language, GPT-4 can solve novel and difficult tasks that span mathematics, coding, vision, medicine, law, psychology and more, without needing any special prompting. Moreover, in all of these tasks, GPT-4's performance is strikingly close to human-level performance, and often vastly surpasses prior models such as ChatGPT. Given the breadth and depth of

2023, arxiv.org/abs/2303.12712

LLMs are VERY easy to customize
This is what makes them "general":
Able to model many different tasks... like Play-Doh?

Sparks of Artificial General Intelligence: Early experiments with GPT-4

Sébastien Bubeck Varun Chandrasekaran Ronen Eldan Johannes Gehrke
Eric Horvitz Ece Kamar Peter Lee Yin Tat Lee Yuanzhi Li Scott Lundberg
Harsha Nori Hamid Palangi Marco Tulio Ribeiro Yi Zhang

Microsoft Research

Abstract

Artificial intelligence (AI) researchers have been developing and refining large language models (LLMs) that exhibit remarkable capabilities across a variety of domains and tasks, challenging our understanding of learning and cognition. The latest model developed by OpenAI, GPT-4 [Ope23], was trained using an unprecedented scale of compute and data. In this paper, we report on our investigation of an early version of GPT-4, when it was still in active development by OpenAI. We contend that (this early version of) GPT-4 is part of a new cohort of LLMs (along with ChatGPT and Google's PaLM for example) that exhibit more general intelligence than previous AI models. We discuss the rising capabilities and implications of these models. We demonstrate that, beyond its mastery of language, GPT-4 can solve novel and difficult tasks that span mathematics, coding, vision, medicine, law, psychology and more, without needing any special prompting. Moreover, in all of these tasks, GPT-4's performance is strikingly close to human-level performance, and often vastly surpasses prior models such as ChatGPT. Given the breadth and depth of

2023, arxiv.org/abs/2303.12712

LLMs enable faster prototyping & iteration

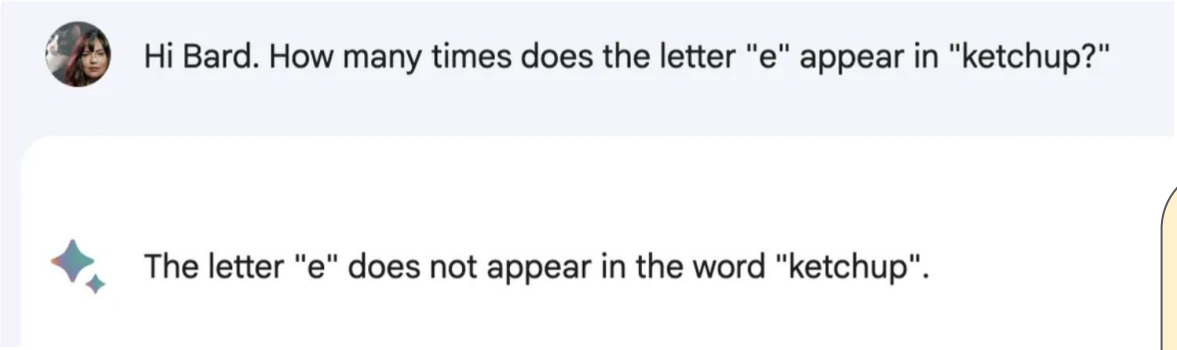
Risk: pressure to release applications prematurely.

But... also represents a key advance for responsible AI development:

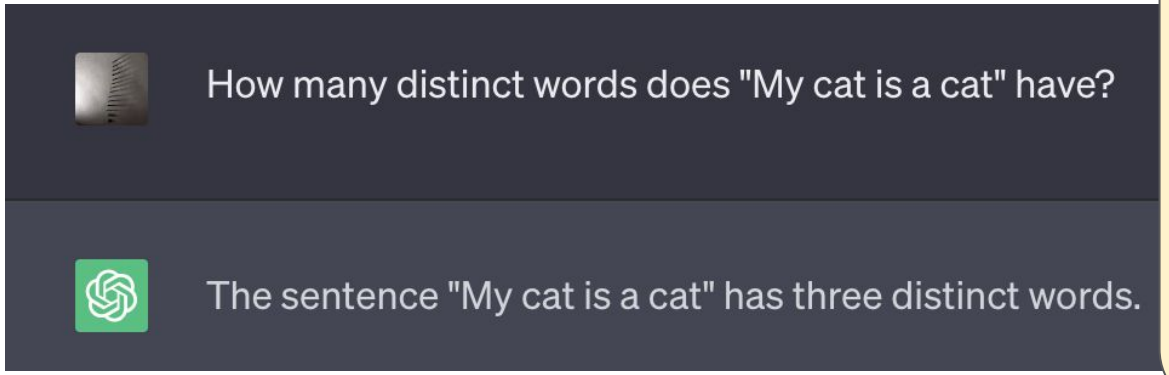
We can iterate much faster with higher fidelity human feedback

[1] PromptMaker: Prompt-based Prototyping with Large Language Models -- ACM CHI 2022

Jagged Frontier (Cat and mouse of showing they are dumb & tuning it out)



Note: tool-use fixes can hide simple examples like this, but does not fix the underlying model's capabilities.



Significant research on the limits of LLMs...

[0] The Illusion of Thinking [Apple] -- https://ml-site.cdn-apple.com/papers/the-illusion-of-thinking.pdf, 2025 *

[1] Faith and Fate: Limits of Transformers on Compositionality -- arxiv:2305.18654, 2023 → Can't multiply

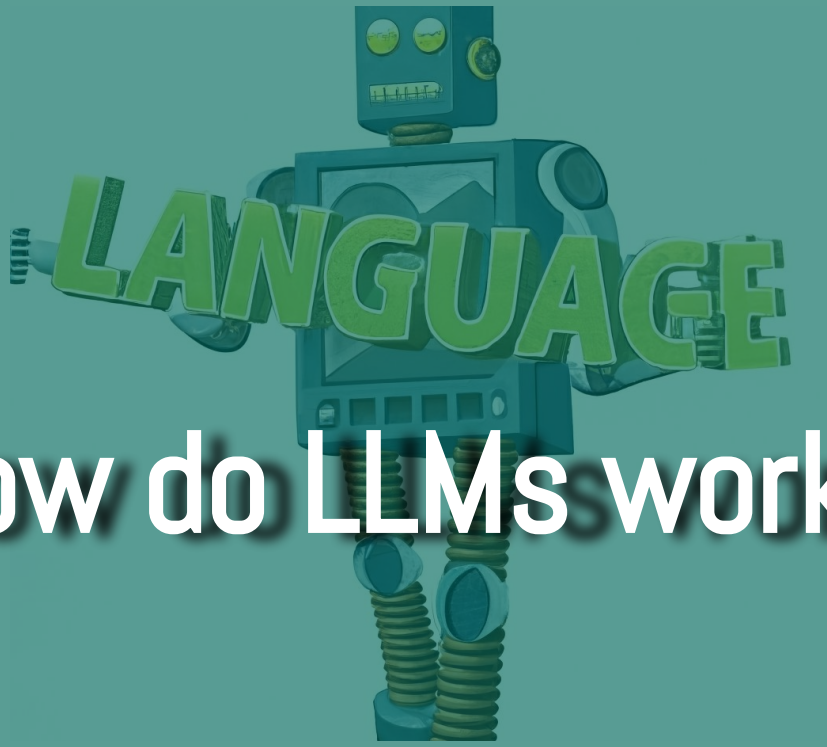
[2] Large Language Models Still Can't Plan -- arxiv:2206.10498, 2023

[3] On the Paradox of Learning to Reason from Data -- arxiv:2205.11502, 2022 → Misgeneration

LLMS are less like a Medium

* chatbots could reasonably be considered a medium

LLMs are more like a Material

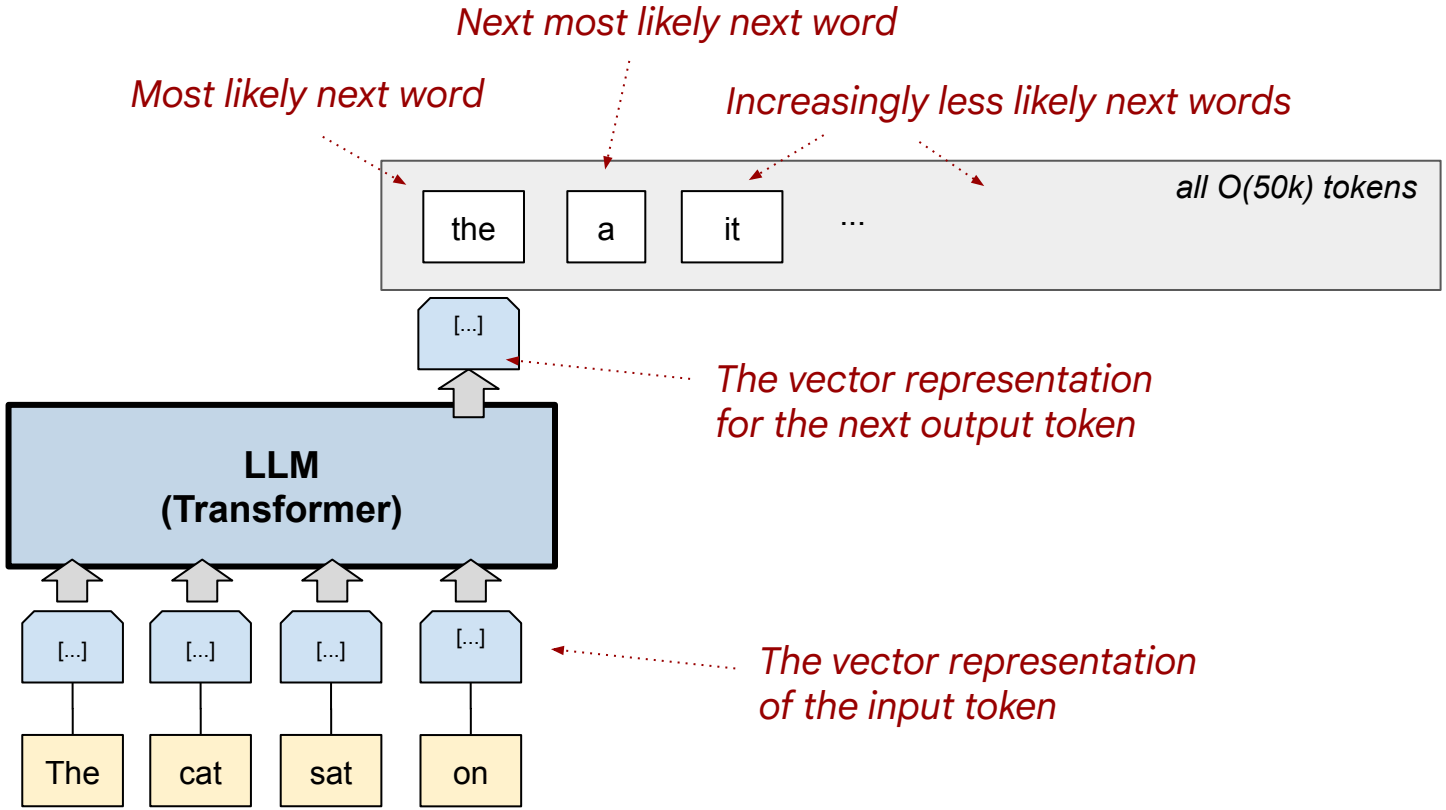


How do LLMs work??

https://arxiv.org/pdf/2303.12712.pdf

What is a Large Language Model?

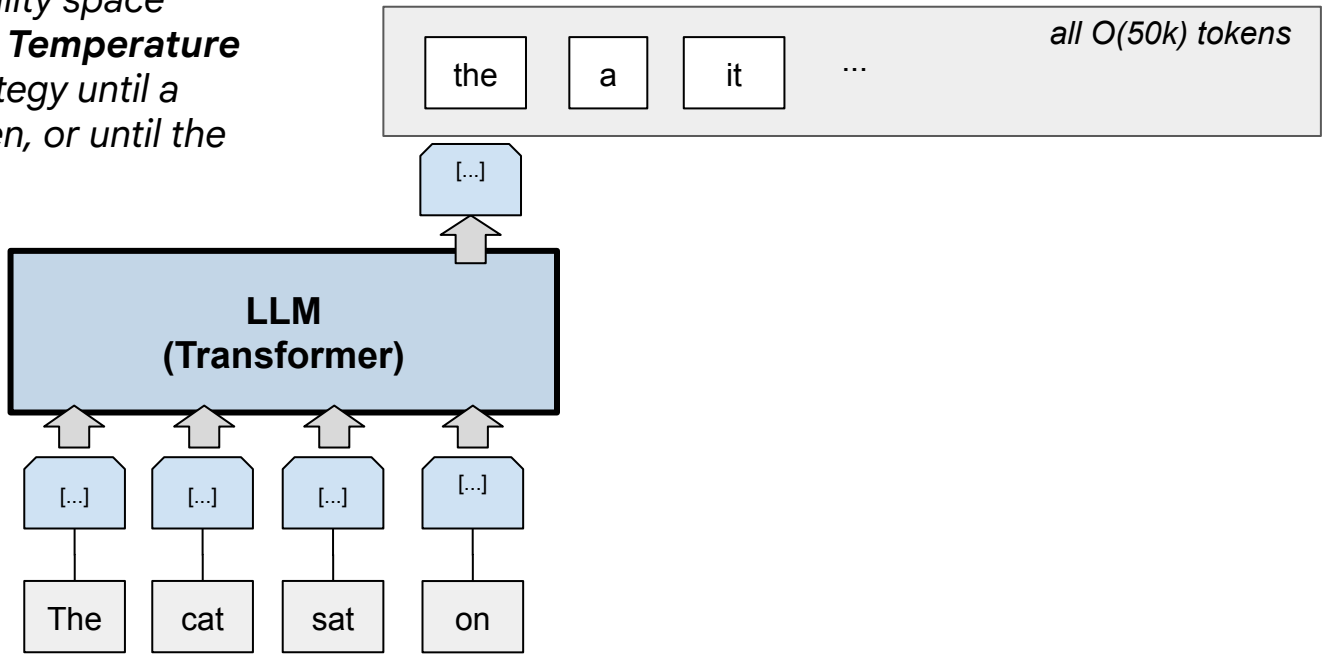
A next token predictor... Deterministic!



token = part of a word, the atomic unit that LLMs work with

What is a Decoding Strategy? Next token "picker"...

<u>Decoding strategy: Sampling</u>
Pick the next word...
*- Only from **top-k** words*
*- Only from **top-p** probability space*
*- Flatten the distribution = **Temperature***
***Repeat** the decoding strategy until a special "EOS" token chosen, or until the max-length is reached*



What is a Decoding Strategy? Next token picker...

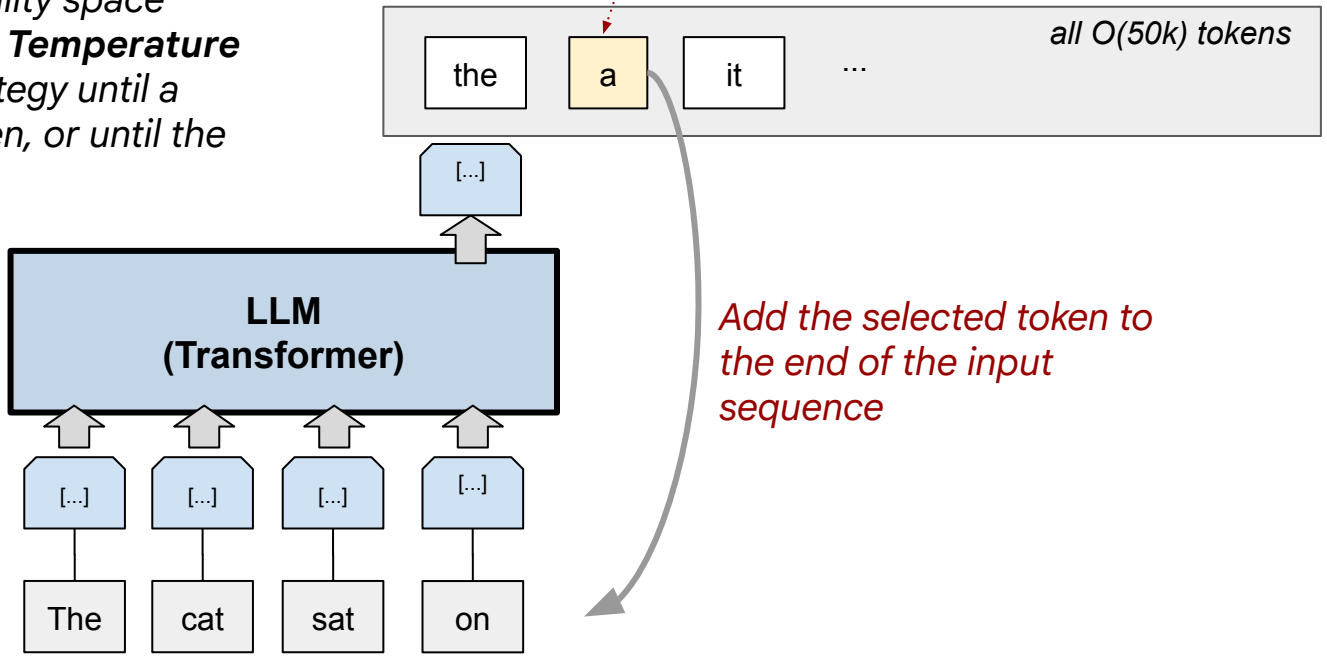
Decoding strategy: Sampling

Pick the next word...

- *Only from **top-k** words*
- *Only from **top-p** probability space*
- *Flatten the distribution = **Temperature***

***Repeat** the decoding strategy until a special "EOS" token chosen, or until the max-length is reached*

Add non-determinism to pick the next word to output...



Add the selected token to the end of the input sequence

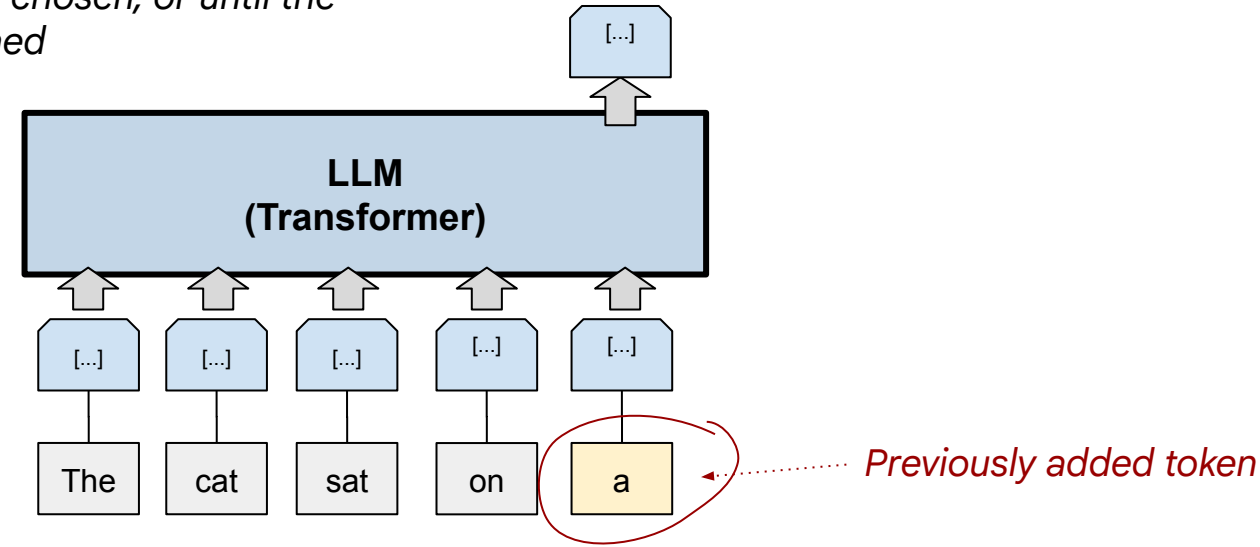
What is a Decoding Strategy? Next token picker...

Decoding strategy: Sampling

Pick the next word...

- *Only from **top-k** words*
- *Only from **top-p** probability space*
- *Flatten the distribution = **Temperature***

***Repeat** the decoding strategy until a special "EOS" token chosen, or until the max-length is reached*



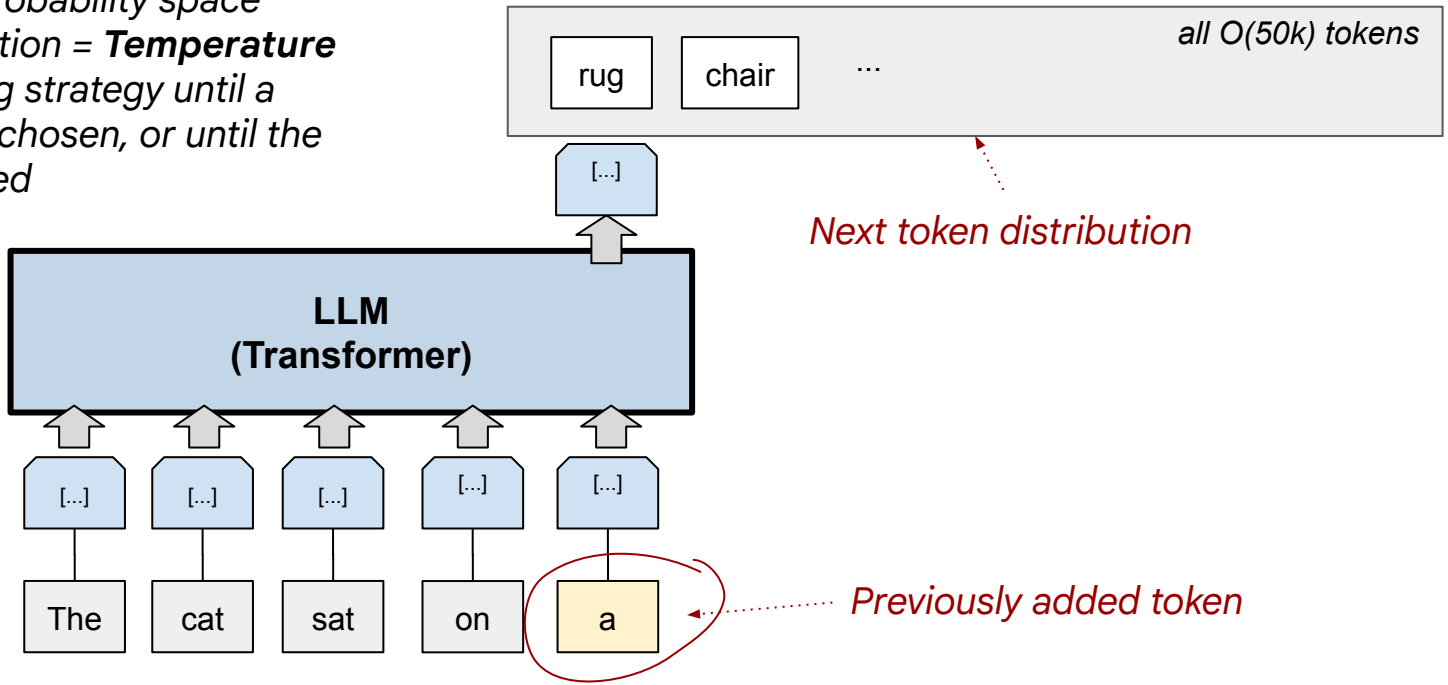
What is a Decoding Strategy? Next token picker...

Decoding strategy: Sampling

Pick the next word...

- *Only from **top-k** words*
- *Only from **top-p** probability space*
- *Flatten the distribution = **Temperature***

***Repeat** the decoding strategy until a special "EOS" token chosen, or until the max-length is reached*



What is a Decoding Strategy? Next token picker...

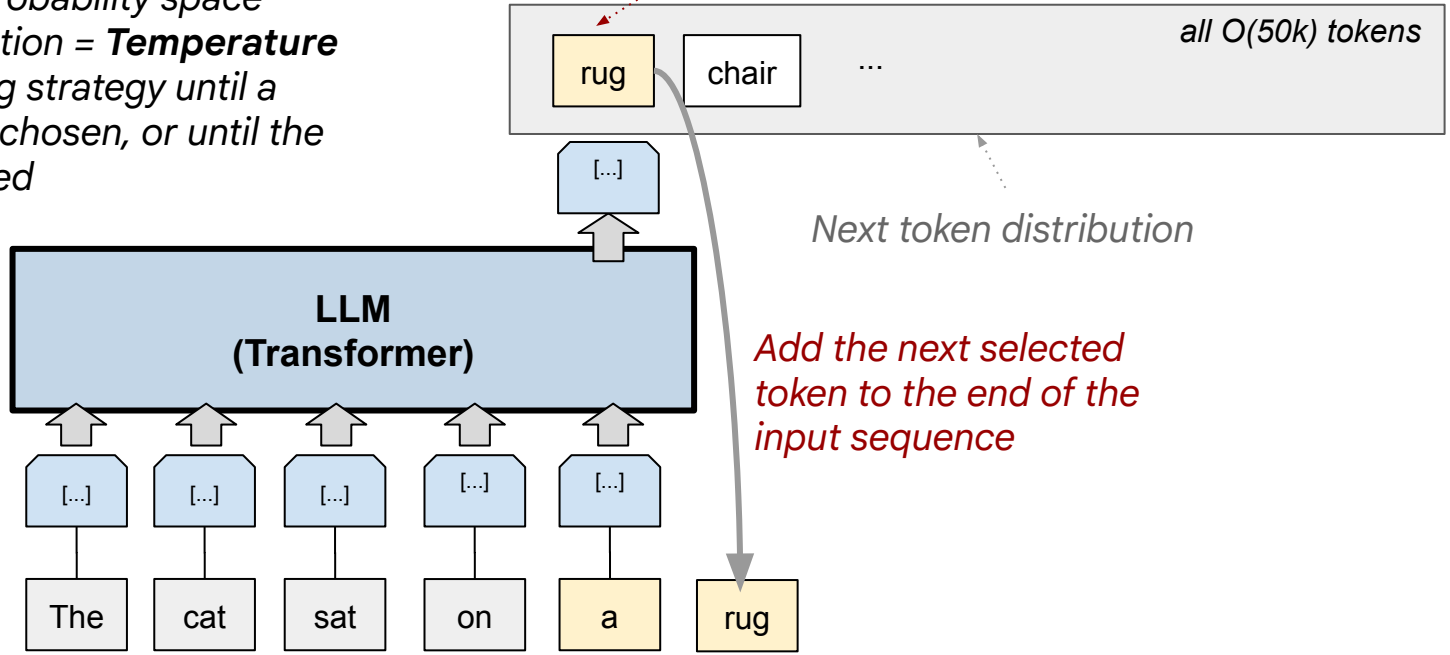
Decoding strategy: Sampling

Pick the next word...

- *Only from **top-k** words*
- *Only from **top-p** probability space*
- *Flatten the distribution = **Temperature***

***Repeat** the decoding strategy until a special "EOS" token chosen, or until the max-length is reached*

Add non-determinism to pick the next word to output...

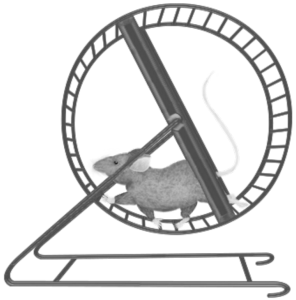


Add the next selected token to the end of the input sequence

Decoding strategies have non-trivial dynamics

e.g. Balancing Temperature

If temperature is **too low**, models tend to gets stuck in repeated loops....



Why this happens is still an open research question, but some insight in [1] SpQR: A Sparse-Quantized Representation for Near-Lossless LLM Weight Compression*, arxiv:2306.03078, 2023*

If it is **too high**, they output more random and less sensible
(& sometimes less true!)

LLMs are VERY adaptable & easy to customize

This is what makes them "general": able to solve so many different tasks... like Play-Doh?

Controlling Large Language Models

Sparks of Artificial General Intelligence: Early experiments with GPT-4

Sébastien Bubeck Varun Chandrasekaran Ronen Eldan Johannes Gehrke
Eric Horvitz Ece Kamar Peter Lee Yin Tat Lee Yuanzhi Li Scott Lundberg
Harsha Nori Hamid Palangi Marco Tulio Ribeiro Yi Zhang

Microsoft Research

Abstract

[illegible] have been developing and refining large language models (LLMs) [illegible] across a variety of domains and tasks, challenging our understanding [illegible] model developed by OpenAI, GPT-4 [Ope23], was trained using an [illegible] data. In this paper, we report on our investigation of an early version of GPT-4, when it was still in a [illegible] development by OpenAI. We contend that (this early version of) GPT-4 is part of a new cohort of LLMs (along with ChatGPT and Google's PaLM for example) that exhibit more general intelligence than previous AI models. We discuss the rising capabilities and implications of these models. We demonstrate that, beyond its mastery of language, GPT-4 can solve novel and difficult tasks that span mathematics, [illegible]ding, [illegible]on, medicine, law, psychology and more, without needing any [illegible] close to human-level [illegible] vastly [illegible] breadth and depth of

Model Control Landscape

Most LLM research and prototyping has been focused here

high training-cost → low training-cost

Pre-training
(Next token)

- ✗ Expensive & slow
- ✗ Incompatible downstream goals

e.g. detoxification of training data damages ability for downstream applications to detect toxicity.
[1] Challenges in Detoxifying Language Models, EMNLP 2021

e.g. GPT3, Palm

Post-training
(Fine tuning all params)

- ✗ Sample expensive
- ✗ Slow iteration
- ✗ Serve N models
- ✓ High quality (& no training example limit)
- ✓ Can use established approaches to responsibility

e.g. RLHF, RLVR

Parameter-Efficient Tuning (tune param subset)

- ✓ Sample efficient
- ✓ Quick(ish) iteration
- ✓ Serve 1 model
- ✓ High quality (& no training example limit)
- ✓ Can use established approaches to responsibility

e.g. Prompt Tuning, IA3, LoRA

Many established ML techniques can be used, e.g. loss functions, influence functions, negative tuning, bias-loss, etc

Prompts
(textual templates)

- ✓ Sample efficient *(easy to work with)*
- ✓ Quick iteration
- ✓ Serve 1 model
- ✗ "Lower" quality (depending on model size)
- ✗ Ad-hoc & few responsibility tools

e.g. unclear what to do about an undesirable output; punctuation, spacing, example ordering all have unpredictable impact on behaviour

e.g. Chain of thought reasoning

"Conventional" Control Wisdom

Most LLM research and prototyping has been focused here

high training-cost → low training-cost

Pre-training

Core Capabilities

- ...nsive & slow
- ...downstream goals

e.g. detoxification of training data damages ability for downstream applications to detect toxicity.
[1] Challenges in Detoxifying Language Models, EMNLP 2021

e.g. GPT3, Palm

Post-training
(Fine Tuning all params)

Follow Instructions to use capabilities

- ...nsive
- ...dels
- ...(& no training example limit)
- Can use established approaches to responsibility

e.g. DExperts, WT5, RLHF

Parameter-Efficient Tuning (PET/PEFT)

Make the LLM do what you actually want

- High quality (& no training example limit)
- Can use established approaches to responsibility

Many established ML techniques can be used, e.g. loss functions, influence functions, negative tuning, bias-loss, etc

e.g. Prompt Tuning, IA3, LoRA

Prompts
(textual templates)

- ...ficient *(easy to work with)*
- Quick iteration
- Serve 1 model
- "Lower" quality (depending on model size)
- Ad-hoc & few responsibility tools

e.g. unclear what to do about an undesirable output; punctuation, spacing, example ordering all have unpredictable impact on behaviour

e.g. Chain of thought reasoning

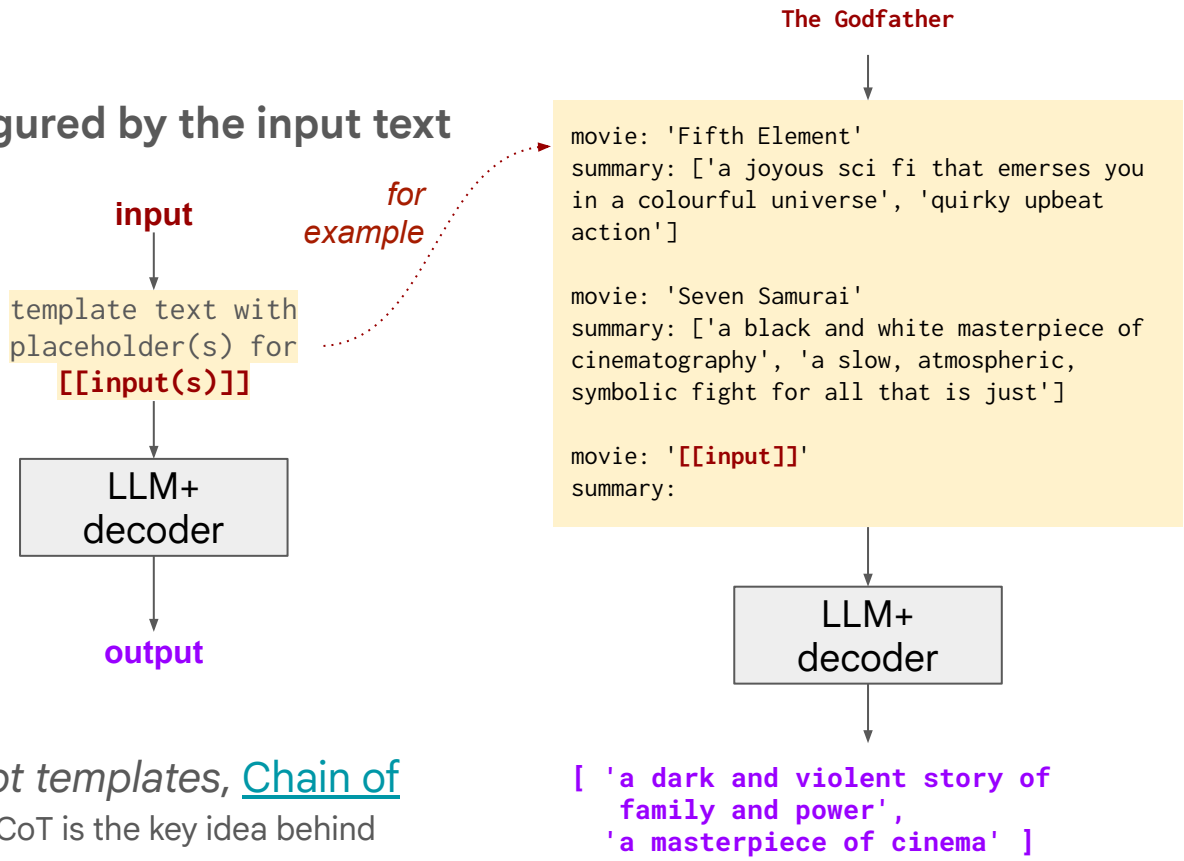
Prompt-Templates

(aka prompt-engineering)

The model behaviour is configured by the input text

& configuration parameters:

- max output length
- stop-tokens
- decoding strategy
- temperature (randomness)
- etc



Many prompting tricks: *few shot templates*, Chain of Thought (CoT) reasoning, etc (CoT is the key idea behind today's thinking models!)

More about prompt-templates

(aka prompt-engineering)

Zero-shot, Instruction, 1 input
(observe: no personalisation)

```
Give a short summary of what
someone might like about the
movie: [[input]]
```

Taxonomy
(for what people do in prompts)

- Instructions
- Few-shot examples
- Explanations/Thoughts
- Structured syntax

Few Shot, JSON syntax, No instruction, 1 input

```
{ movie: 'Fifth Element'
  summary: ['a joyous sci fi
that emerses you in a
colourful universe', 'quirky
upbeat action'] }

{ movie: 'Seven Samurai'
  summary: ['a black and
white masterpiece of
cinematography', 'a slow,
atmospheric, symbolic fight
for all that is just'] }

{ movie: '[[input]]'
  summary:
```

"Chain of Thought" instruction (ask for explanation), 2 input

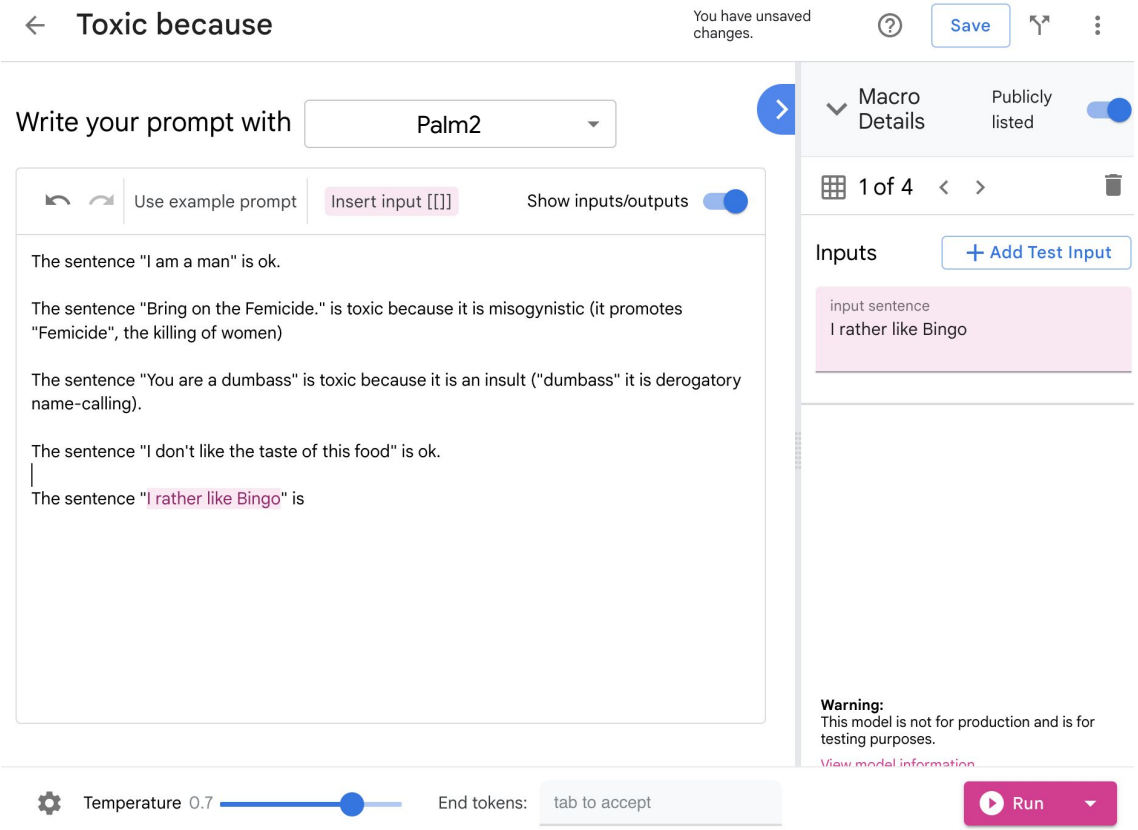
```
Given the following
preferences, and a movie,
describe the movie in terms
of the preferences, and then
conclude with whether the
movie will be liked or not.

Preferences: [[preferences]]
Movie: [[input]]
Description:
```

A toxicity classifier in 5 minutes...

Before... [In 2016-2017]
2m comments x 10 annotations
+ 1yr development & ML experts
→ Perspective API
[www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge]

In 2023
80 examples, 1 annotator + LLM
[1] "Towards Agile Text Classifiers for Everyone" 2023 [arxiv.org:2302.06541]

& LLM+PET won ACL SemEval competition to identify & explain sexist language
[2] SemEval-2023 Task 10: Explainable Detection of Online Sexism, 2023 [arxiv.org:2303.04222]



[3] PromptMaker: Prompt-based Prototyping with Large Language Models -- ACM CHI 2022

Modern Prompting: system, developer, user instructions & function calling

(& decoding controls)

System Instructions

```
Be Helpful, Harmless and
Honest
```

Developer Instructions

```
Summarize what the user
might like about the given
movie.
```

User Input

```
Movie: [[input]]
Preferences:
[[preferences]]
```

Decoding Controls

```
Temperature: 1.0
Constraint: {
  Like: "<'yes'|'no'>"
  Why: "<string>"
}
...
```

Combined Template + Input

```
<system> Helpful, ...
<dev> Summarize ...
<user> Movie:The Godfather...
<function-cal> ...
<output>
```

Special output tokens, usually learnt in post-training

"Tools" (e.g. google search, run code, etc)

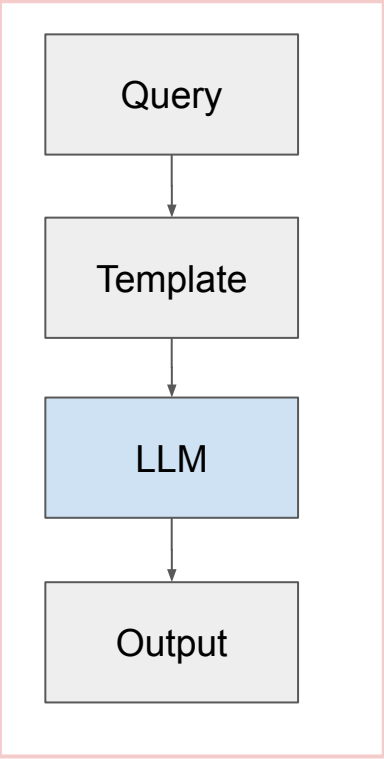
AI System

LLM

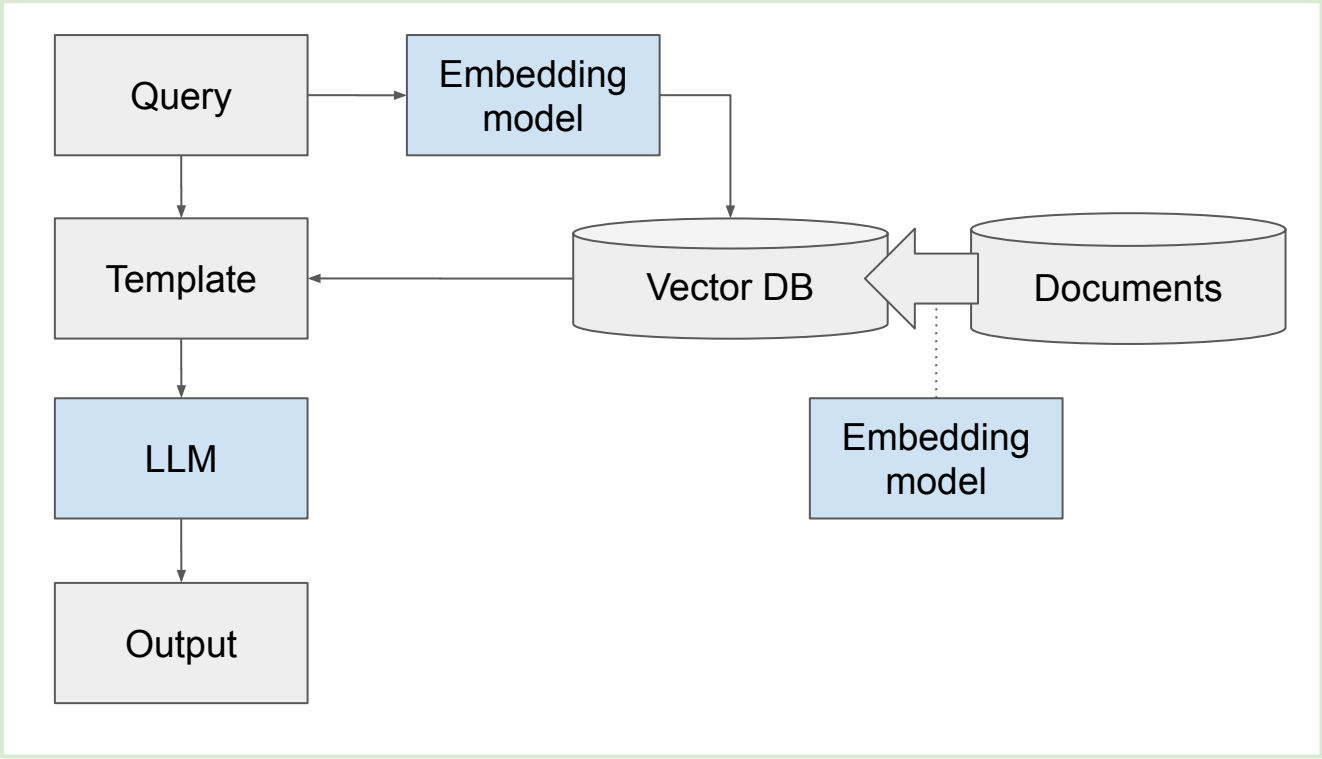
```
...final output...
<eos>
```

Modern

Prompting



Retrieval Augmented Generation



Context Engineering

How do you get the right stuff into a models' context?

- Some models have very large contexts (e.g. Gemini has 1m tokens)
- But: models are slow, and still struggle if the context is too big or contains distracting irrelevant information
- Context engineering
  - Engineering to get the right information into a model's context
  - RAG is one tool in the Context Engineering toolbox
  - "Tables of Contents" + multiple queries is another
    - e.g. file list + summary to find the right files to load when coding

Agents (and prompt-chains)

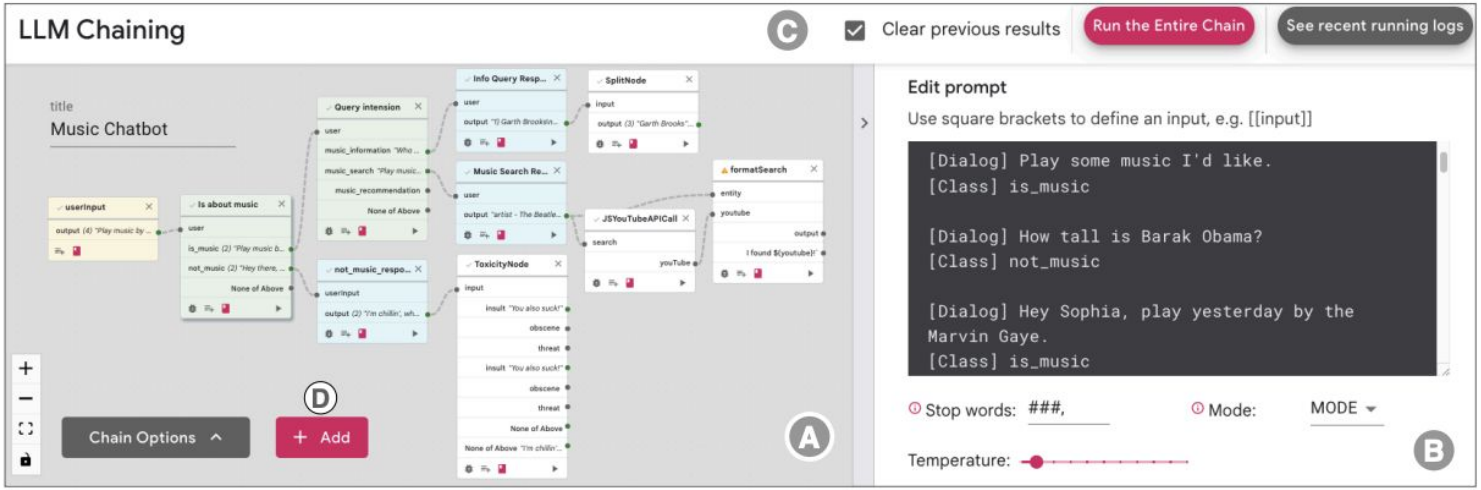


Figure 1: The PromptChainer interface. (A) The *Chain View* visualizes the chain structure with node-edge diagrams (enlarged in Figure 2), and allows users to edit the chain by adding, removing, or reconnecting nodes. (B) The *Node View* supports implementing, improving, and testing each individual node, *e.g.*, editing prompts for LLM nodes. PromptChainer also supports running the chain end-to-end (C).

"PromptChainer: Chaining Large Language Model Prompts through Visual Programming"
-- Tongshuang Wu, Ellen Jiang, Aaron Donsbach, Jeff Gray, Alejandra Molina, Michael Terry, Carrie J Cai

"Orchestrating AI" is very active of research

LangChain, LangGraph, Haystack, ReAct, etc...

Inference for Quality: v*alue functions...*

Scaling inference to get quality requires...
Knowing what you want

- Sampling (naive)
  - Imagine 1/100 success rate; but "you know if it's right when you see it"...
  - 99.99% success rate: just sample 1k, and test each

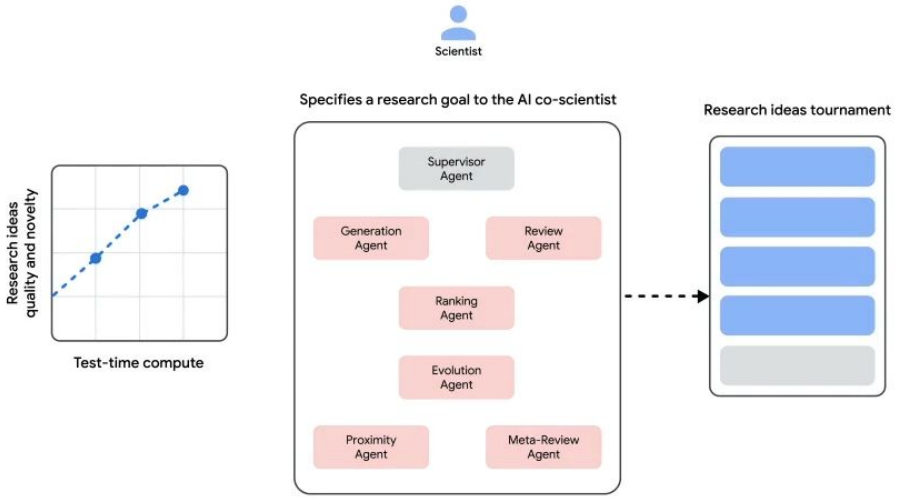
- Sample & Iterate with an LLM
  Genetic programming with LLMs as mutation

  - **Absolute ranking** (typically with Custom measurement code)

  - **Pairwise ranking** (e.g. ELO rating): allows LLMs themselves to rate/score pairs of examples

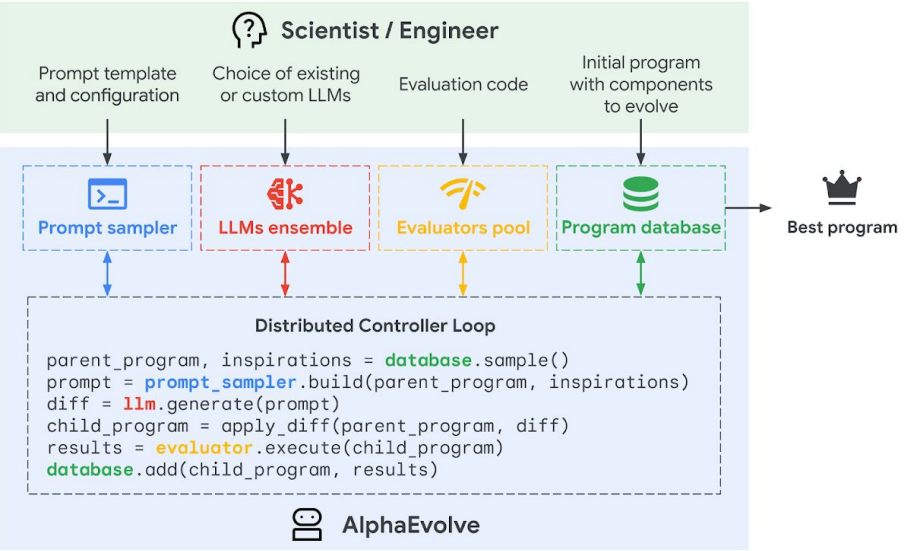
Inference for Quality: Examples

[1] "Accelerating scientific breakthroughs with an AI co-scientist"



Suggested applications are now being explored by ICL;
Generated hypotheses and experimental protocols for target discovery hypotheses, focusing on liver fibrosis, being explored by researchers at Stanford University.

[2] "AlphaEvolve: A Gemini-powered coding agent for designing advanced algorithms"



AlphaEvolve achieved up to a 32.5% speedup for the FlashAttention; 1% improvement on Gemini training time; improved 4x4 matrix multiplication...

[1] https://research.google/blog/accelerating-scientific-breakthroughs-with-an-ai-co-scientist/
[2] https://deepmind.google/discover/blog/alphaevolve-a-gemini-powered-coding-agent-for-designing-advanced-algorithms/

Prototypes & Responsibility

LLMs enable faster prototyping & iteration
Risk pressure to release applications prematurely.
But... also represents a key advance for responsible AI development:
We can iterate much faster with more real human feedback

[1] PromptMaker: Prompt-based Prototyping with Large Language Models -- ACM CHI 2022

Responsible AI

Make AI with ethical principles and societal values

Responsible AI Make AI with ethical principles and societal values

Mitigate world-negative impact

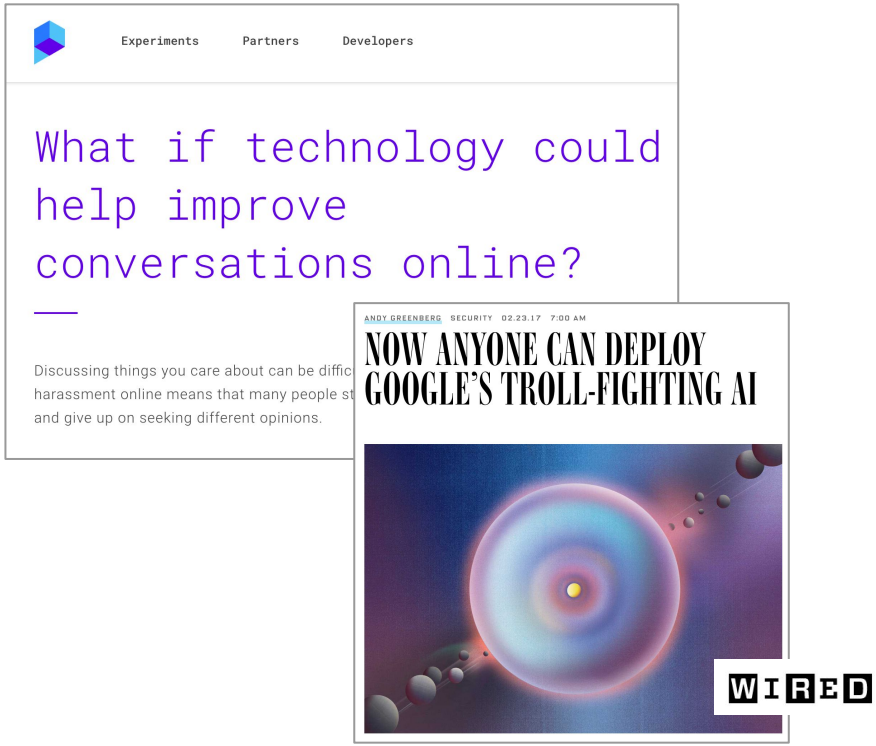
- Fairness: biases & inequalities?
  - Definitions are incompatible
  - Reflect reality vs Creator's ideals? (simulation, mitigate toxicity, vs maybe fairer?)
- Transparency & Explainability: trust?
  - While interpretability methods have made a lot of progress, abstractions are necessarily lossy...
- Privacy & Security
  - Extra complication: training can memorize
- Safety & Alignment
  - Intent clash: System vs developer vs user
  - Misinformation?
- Environmental concerns

Enable world-positive impact

- Science
  - AlphaFold & New drug discovery
  - AlphaEvolve & Computer Science
  - AlphaProof & Mathematics
- Health
  - MedGemma: for medical text and image comprehension
- Democratise software
  - Automate bureaucracy
- Education
  - Anyone can learn anything far easier than before? (e.g. LearnLM)
- Misinformation & polarization?
  - Chatbots reduce the spread of misinformation -- Pennycook, Rand, 2022

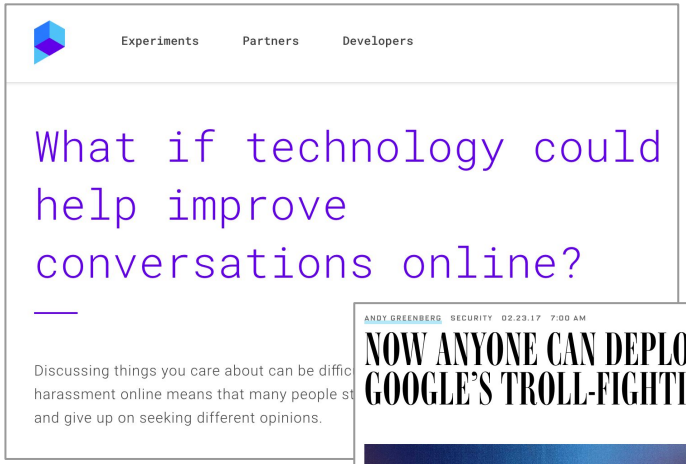
2017... Perspective API

(CNN to detect toxicity online)



2017... Perspective API

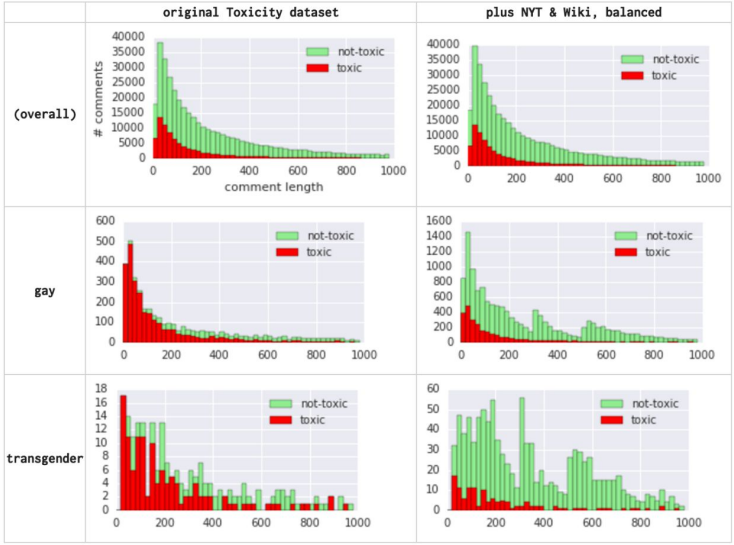
(CNN to detect toxicity online)



2017... False positives - balancing datasets to tackle bias

Comment	Old	**New**
The Gay and Lesbian Film Festival starts today.	82%	**1%**
Being transgender is independent of sexual orientation.	52%	**5%**
A Muslim is someone who follows or practices Islam.	46%	**13%**

No significant quality change in ROC-AUC



[Measuring and Mitigating Unintended Bias in Text Classification](#) -- Dixon et al. AIES'2018

Unintended biases can have unexpected impacts

negative... *or positive!*

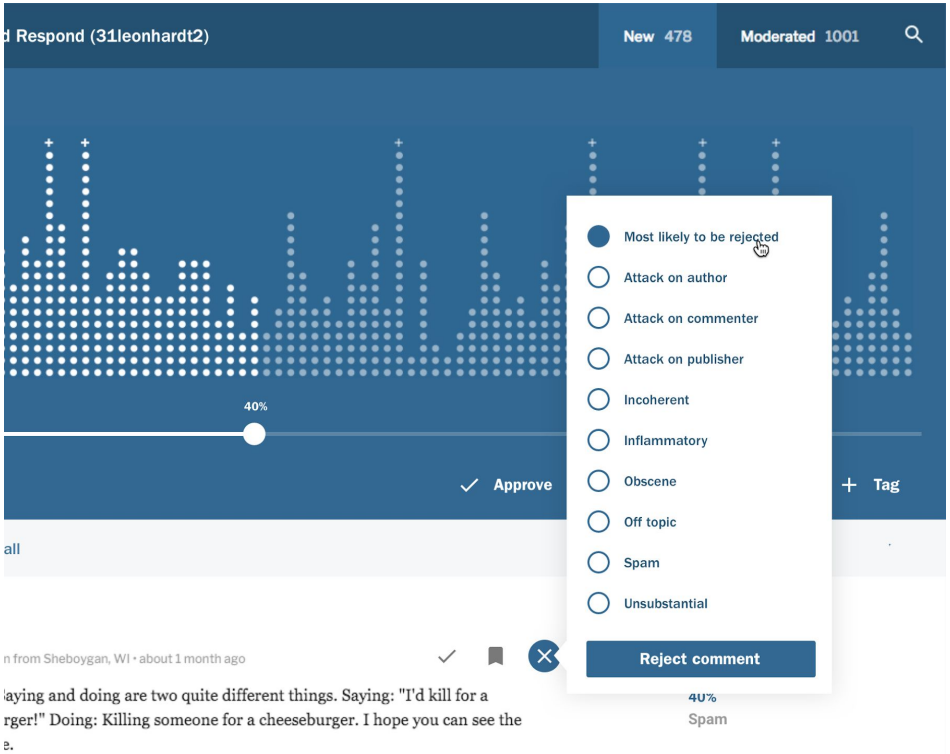
2017... NYT moderation
(with Perspective API):

1. Pre-review all comments
2. Sort by most-toxic first
3. After each comment is reviewed, if accepted, publish immediately.

Positive comments containing the word "gay" then get ***more*** human-attention than before.

But... opposite if the sort order was reversed.

UI choice inverts model fairness impact!



https://github.com/conversationai/conversationai-moderator

LLMs have unintended biases too...

but... **the biases are very customizable!**

e.g. And much less than 2017's AI; e.g. no special configuration:
"5 minute toxicity classifier":

The Gay and Lesbian Film Festival starts today.	ok
Being transgender is independent of sexual orientation.	ok
A Muslim is someone who follows or practices Islam.	ok

LLMs' ability to understand language is far better than 2017 models (Bert)

Reflection is needed: *you must choose & study your prompt (= your bias)*
("jailbreaking LLMs": a model's "default" biases can be tricked/changed, even if we don't want it to!)

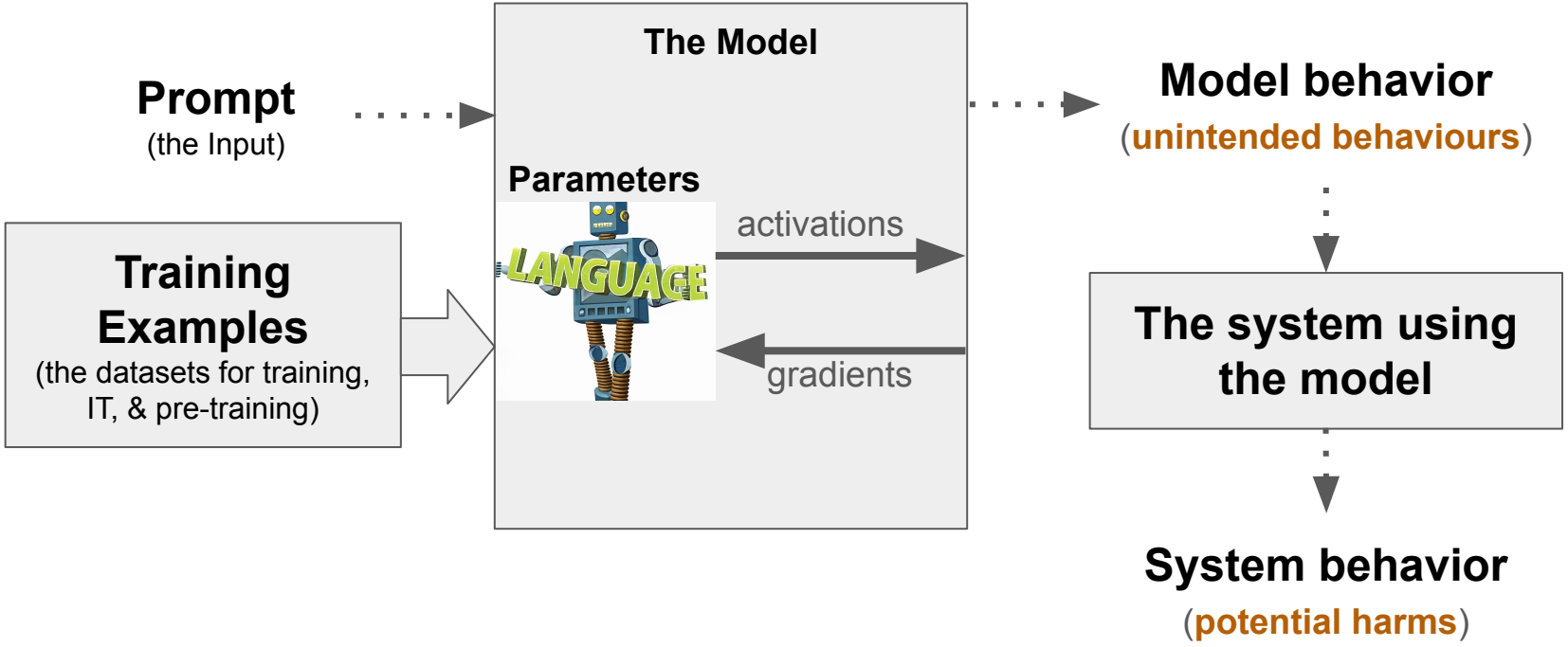
LLM Interpretability:
Understanding the Material

LLMS are less like a Medium

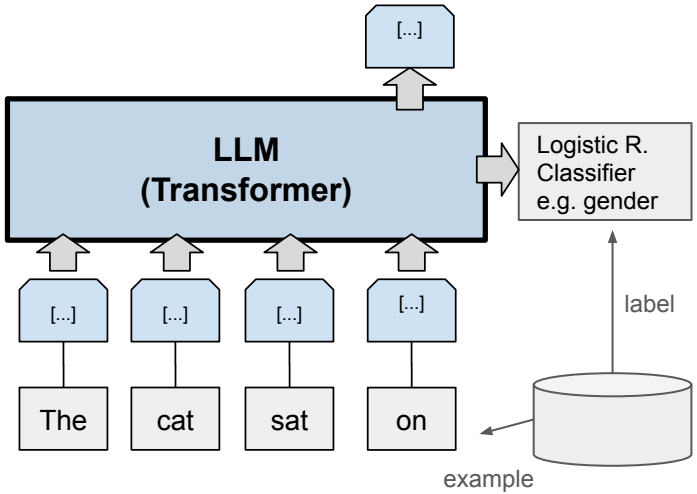
LLMs are more like a Material

https://arxiv.org/pdf/2303.12712.pdf

Understanding *in terms of control*...



Probes



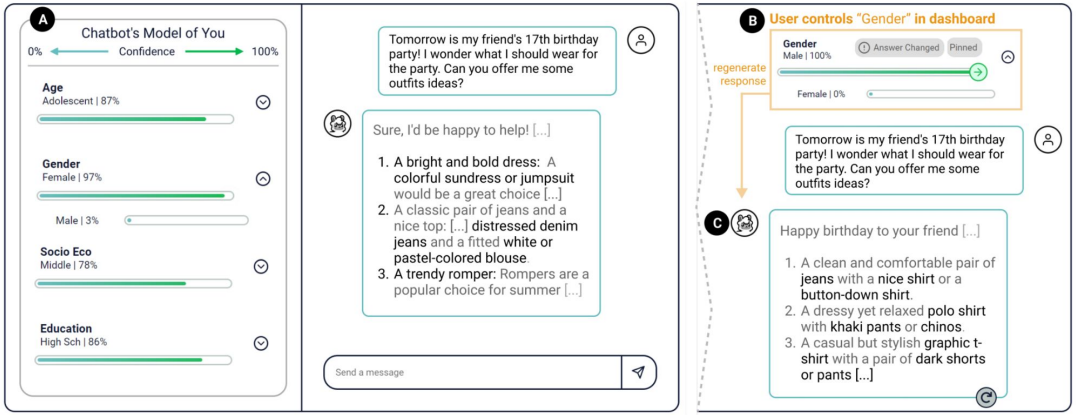


Figure 2: Dashboard interface. (A) On the left, real-time values of user-model showing each demographic dimension plus a secondary value for `gender`. (B) The user modifies "`Gender`" dimension by pinning down "`Male`." (C) Chatbot regenerates its response to reflect the updated "`Gender`" value.

Designing a dashboard for transparency and control of conversational AI -- https://arxiv.org/abs/2406.07882 Yida et al. 2023

- *Concept Activation Vectors (T)CAV: Probes for concepts you care about*
- *Sparse Autoencoders (SAEs): learn probes in an unsupervised way (dense layer = projection down of larger but sparse layer), hot topic in interpretability today*

Interactive Prompt Debugging with **Sequence Salience** (Norm of Gradients): https://arxiv.org/abs/2404.07498 ()

https://github.com/PAIR-code/lit

Datapoint Editor

source CategoryLabel: fewshot-mistake

***prompt** TextSegment

```
Taste-dislikes: Don't like onions or garlic
Suggestion: Onion soup
Analysis: it has cooked onions in it, which you don't like.
Recommendation: You have to try it.

Taste-likes: I've a sweet-tooth
Taste-dislikes: Don't like onions or garlic
Suggestion: Baguette maison au levain
Analysis: Home-made leaven bread in france is usually great
Recommendation: Likely good.

Taste-likes: I've a sweet-tooth
Taste-dislikes: Don't like onions or garlic
Suggestion: Macaron in france
Analysis: Sweet with many kinds of flavours
Recommendation: You have to try it.

## Now analyze one more example:

Taste-likes: Cheese
Taste-dislikes: Can't eat eggs
Suggestion: Quiche Lorraine
Analysis:
```

target TextSegment

```
Contains eggs, not a good option.
```

Add | Add and compare | Reset | Clear

LM Salience

Sequence (response): A savory tart with cheese and eggs Recommendation: You might not like it, but it's worth tryi... | Select sequence

Granularity: Tokens | Words | Sentences | Lines | ¶ Method: grad_l2 | grad_dot_input

```
Taste-likes: I've a sweet-tooth
Taste-dislikes: Don't like onions or garlic
Suggestion: Onion soup
Analysis: it has cooked onions in it, which you don't like.
Recommendation: You have to try it.

Taste-likes: I've a sweet-tooth
Taste-dislikes: Don't like onions or garlic
Suggestion: Baguette maison au levain
Analysis: Home-made leaven bread in france is usually great
Recommendation: Likely good.

Taste-likes: I've a sweet-tooth
Taste-dislikes: Don't like onions or garlic
Suggestion: Macaron in france
Analysis: Sweet with many kinds of flavours
Recommendation: You have to try it.

## Now analyze one more example:

Taste-likes: Cheese
Taste-dislikes: Can't eat eggs
Suggestion: Quiche Lorraine
Analysis: A savory tart with cheese and eggs
Recommendation: You might not like it, but it's worth trying.
```

This one looks important

And we found a mistake in the input!

3 Select target to explain

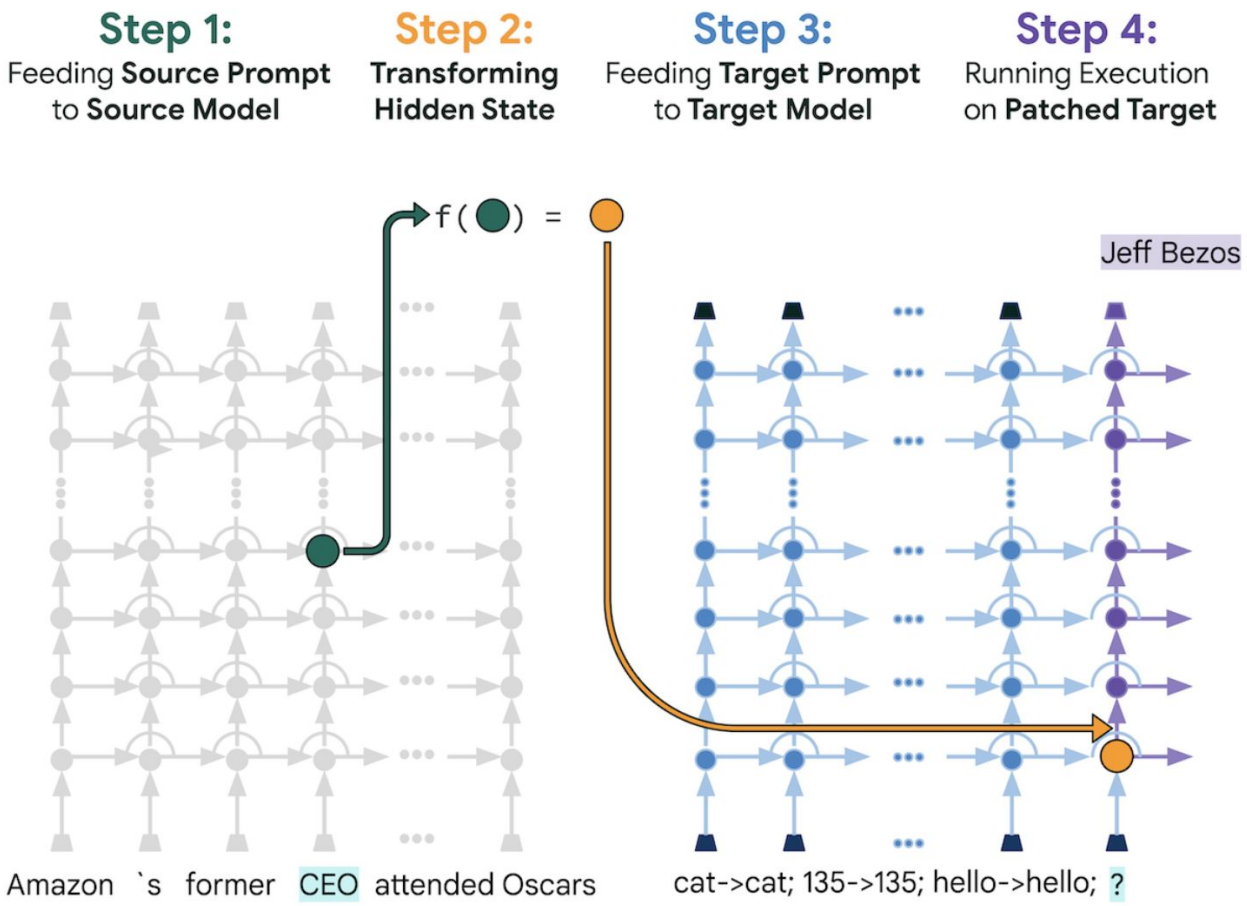
Explaining [183:231] "Tast... Salience 0 1126.03 Colormap intensity: 0 6 1

Learn more about saliency: https://pair.withgoogle.com/explorables/saliency/

Patching

Patchscopes: A Unifying Framework for Inspecting Hidden Representations of Language Models

https://arxiv.org/abs/2401.06102



Patchscopes

Source Prompt: “Alexander the Great”

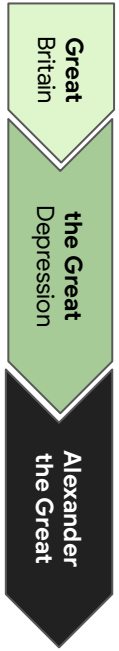
Target Prompt: “Syria: Country in the Middle East, Leonardo DiCaprio: American actor, Samsung: South Korean multinational major appliance and consumer electronics corporation, x”

Source Position: Last

Target Position: Last

Source Layer: Variable

Target Layer: Same as source layer



Source Layer	Generation	"Meaning" context
1	Britain: Country in the European Union	“Great”
2	Wall Street Crash of 1929: Financial crisis in the United States	“the Great”
3	Wall Street Bubble: The Great Depression	“the Great”
4	Wall Street: Wall Street in New York City	“the Great”
5	: Ancient Greek ruler, and the first to rule all of the then known world	“Alexander the Great”

Pythia 12B

Training Data Attribution (TDA) for PETs & Data Cleaning

TDA aims to identify training examples that are the cause a given model's output.

This can be used to clean datasets...

1. Introduce noise into a dataset

2. Use TDA to find train-set-examples that most disagree with the validation set. Throw them away, or relabel.

3. Observe significant performance increase.

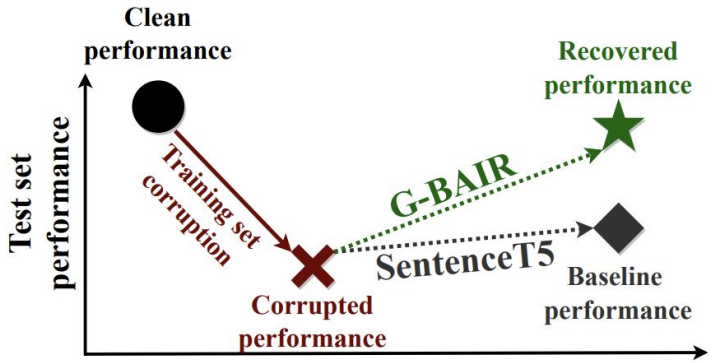
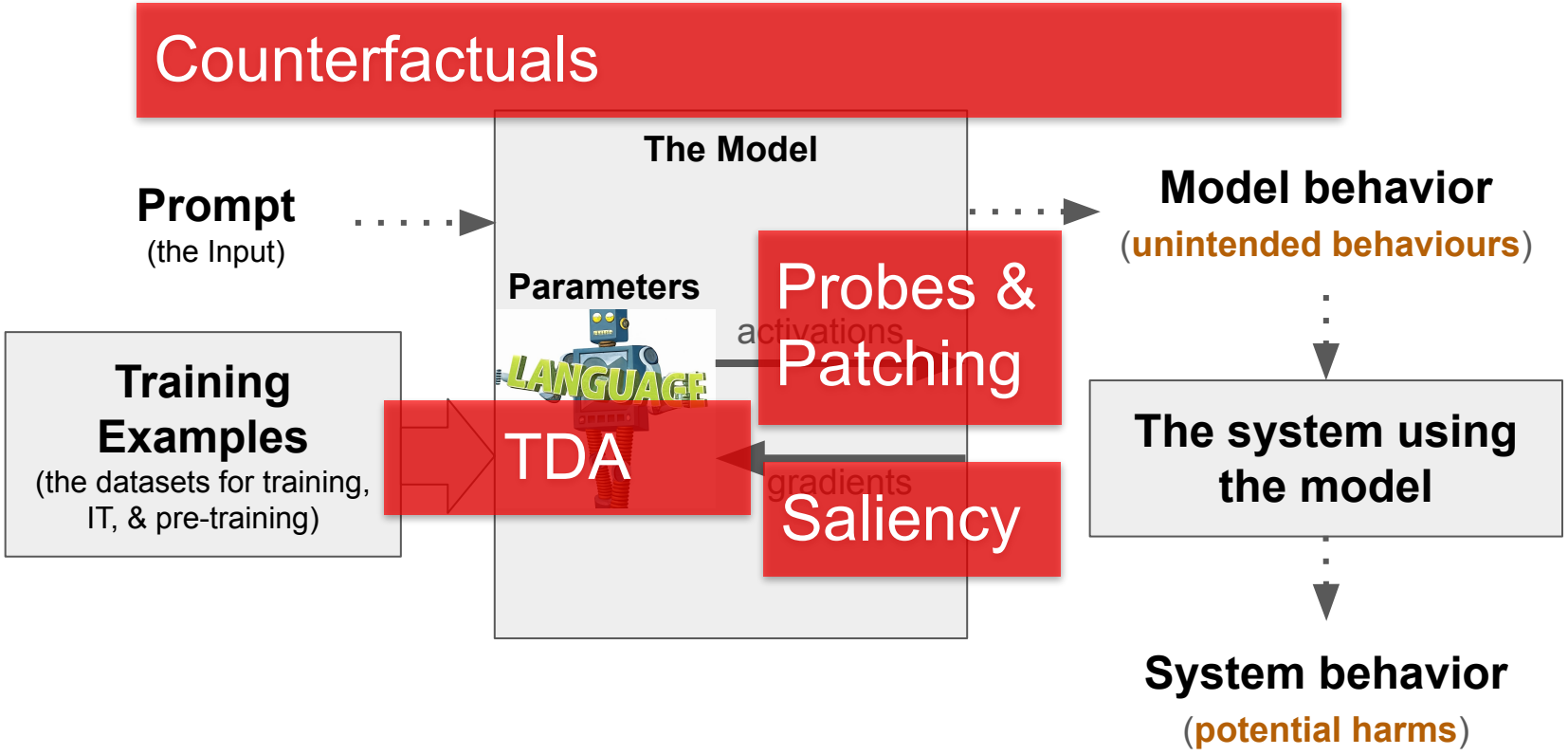


Figure 1: Illustration of our **G-BAIR** method used to recover prompt-tuning model performance drops incurred through data corruption. Clean model performance (●) drops as a result of training data corruption (✖). **G-BAIR** (★) can be applied to identify and mitigate corrupted examples, thereby recovering clean test set performance better than the compared **SentenceT5** (◆) baseline.

[1] Gradient-Based Automated Iterative Recovery for Parameter-Efficient Tuning -- arxiv/2302.06598

Understanding via interpretability methods



Large Language Models:
Building with the Material

LLMS are less like a Medium

LLMs are more like a Material

https://arxiv.org/pdf/2303.12712.pdf

What do people build with LLMs?

- Learning & teaching (and cheating...)
  - What do I need to learn to be able to do X? explain context of a phrase, correct my phrase, etc
  - Give me an exercise for learning X?
- Writing assistance
  - Summarize, style-editing, make it formal, write it appropriately to send to my doctor
- Administrative work
  - Pull all the costs from these PDFs in that spreadsheet
- Coding Tools
  - Auto-complete, make an examples to do X, write code to do Z, what's wrong with my code W, ...
- Marketing & Creative Assistance
  - Making images and video
- Fun
  - Dynamic games, Visualization, "companionship" chatbots, making memes, ...





What do people build with LLMs?

- Learning & teaching (and cheating...)
  - What do I need to learn to be able to do X? explain context of a phrase, correct my phrase, etc
  - Give me an exercise for learning X?
- Writing assistance
  - Summarize, style-editing, make it formal, write it appropriately to send to my doctor
- Administrative work
  - Pull all the costs from these PDFs in that spreadsheet
- **Coding Tools**
  - Auto-complete, make an examples to do X, write code to do Z, what's wrong with my code W, ...
- Marketing & Creative Assistance
  - Making images and video
- Fun
  - Dynamic games, Visualization, "companionship" chatbots, making memes, ...





I want to make a game called "Throwing Rocks at Orks", where the hero is a knight at the top of a tower, and orks climb up and you have to throw rocks onto the orks before they get to the top.

Show thinking

Okay, this sounds like a fun game! I can help you create "Throwing Rocks at Orks." We'll build it using HTML, CSS, and JavaScript, drawing the game on an HTML canvas.

Here's what the game will look like:

- You'll control a knight at the top of a tower using your mouse.
- Orks will start climbing from the bottom of the screen.
- You'll click to make the knight drop a rock onto the orks below.
- If a rock hits an ork, you score a point!

Ask Gemini

Canvas Search

Throwing Rocks at Orks Game

Code Preview Share

Vibe Coding = telling AI to write code for you

- Anyone can do it... but...

  the more you know, the more you can do...

- Probably the most advanced and invested area in LLM applications

Vibe Coding & The (busy) Landscape of AI-Coding

Codey ⟵⟶ **Vibey**

Close to existing human workflows — Autonomous

AI+IDEs
(e.g. Windsurf, Cursor, **RooCode**, GitHub Copilot, Google Code Assists)

Cloud-Integrated Editor
(e.g. Firebase Studio,)

"AI Studios" / Applets
(e.g. Google AI Studio, Amazon sagemaker)

Chatbots that code inline apps
(Gemini & ChatGPT Canvas, Anthropic Artefacts)

Autonomous AI Engineers
(Devin, factory.ai)

Command Line Interface Agents
(e.g. Claude Code, Gemini CLI)

Zero-code solutions
(Lovable, Replit, Bubble)

Mockups to UI Code
Onlook, Galileo AI, v0, Uizard, Figma

Andrej Karpathy: Software Is Changing (Again)
https://www.youtube.com/watch?v=LCEmiRjPEtQ

Vibe Code: Make an interactive visualization of this...

Interactive Elliptic Curve Torsion Visualization

Controls

Family:

Silverman (Fig 6.1)

Section (P):

P = (0,0)

Max Iterations (N): 8

Fast Escape — Slow Escape

Resolution:

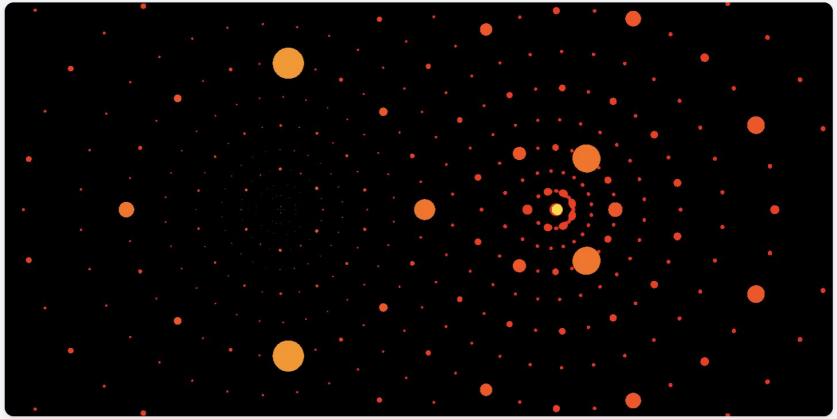
High (Slow)

Escape Threshold (Magnitude):

100

Viewport

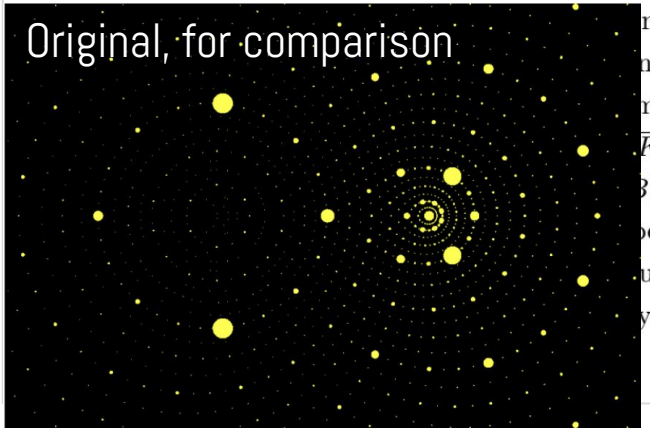
Zoom In



VARIATION OF CANONICAL HEIGH

LAURA DEMARCO AND NIKI

ABSTRACT. Let $\pi : E \to B$ be an elliptic surface d
is a smooth projective curve, and let $P : B \to E$ be
height $\hat{h}_E(P) \neq 0$. In this article, we show that th
ne bund
n and re
ns (of C
$\overline{K}$) whe
$B(\mathbb{C})$. F
pecial se
urves –
y points



Example thank to Martin Wattenberg

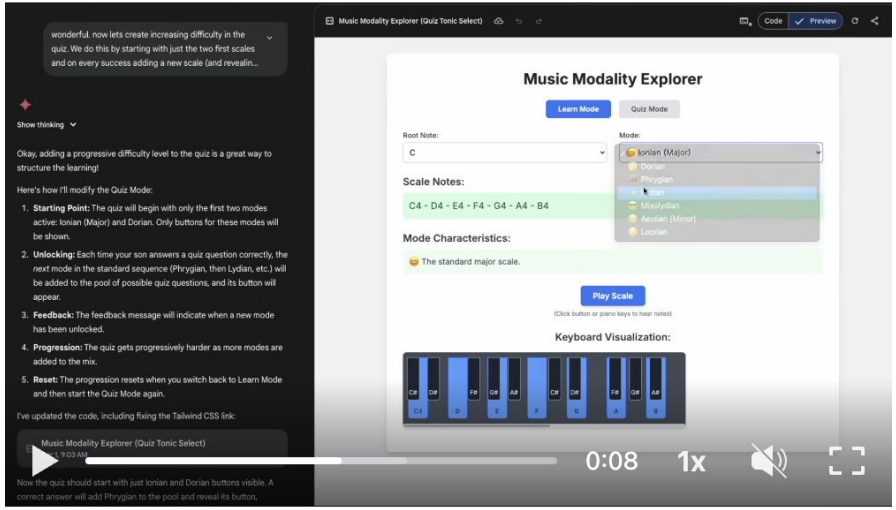
Amit Pitaru • 2nd
Director, Creative Technology x AI, Google Creative Lab. F...
2w •

I was hanging with my son who is now learning music theory, quickly prompted this mini trainer app for him to practice scale modes/modalities

I'm loving this new instant-mini-learning-tools situation were trending towards

Try it out here https://lnkd.in/eYdgE7mY
(source code also included in the link)

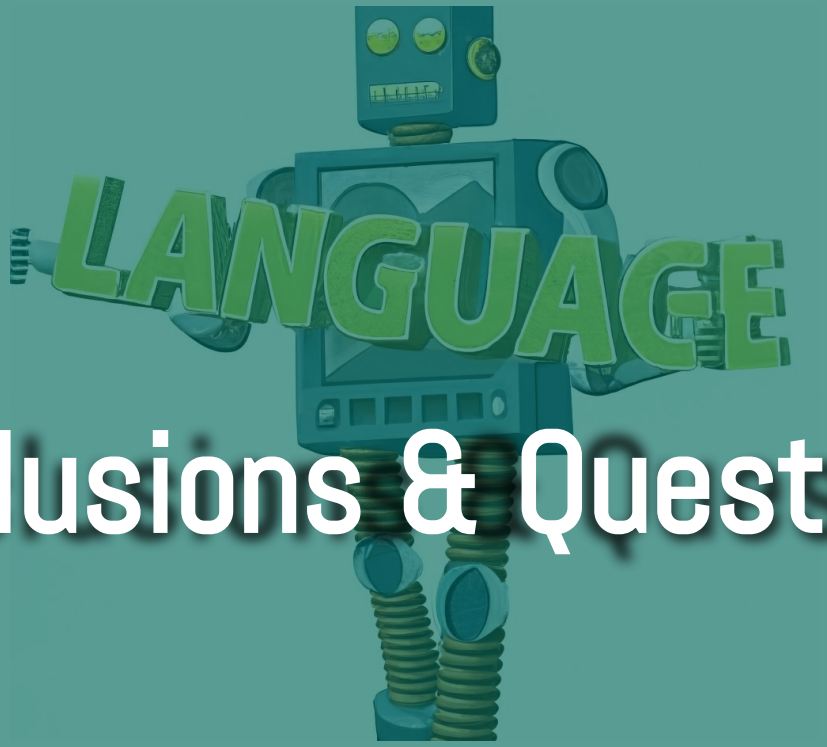
Below is the original session with Gemini from a couple weeks back w/ app already generated



Vibe Coding

~ managing an AI to write code for you

- **Work = "code" → "specification"** (& cost management)
  - Do you know what you want? no...
  - Do you know how to learn what you want?
- **Anyone can vibe-code... but knowledge helps!**
  - Good mental models of how things fit together & actually run (ai/dev-ops!) hugely increases what you can do.
  - Anecdote: some MSc level teachers find bigger differences in student accomplishments
- **Used to learn, teach, for fun & for admin**
- **Advancing very fast** (one of the most active areas of research)...
  - Every couple of months you can do significantly more than the previous month



Conclusions & Questions

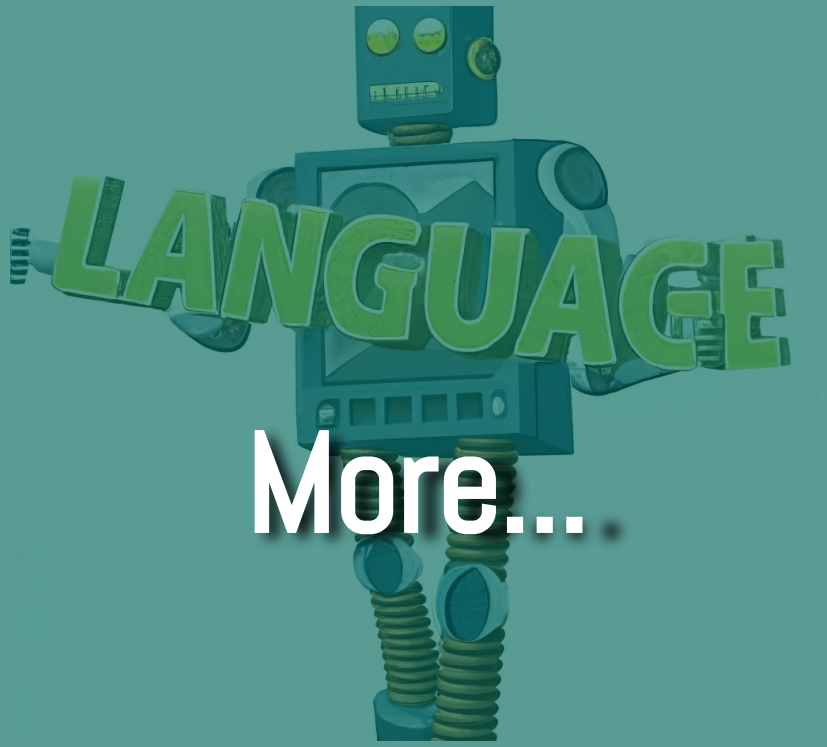
https://arxiv.org/pdf/2303.12712.pdf

Conclusions

- **LLMs: an amazingly versatile new material**
  - Easily customizable (textual-prompts & small datasets)
  - Natural language becomes the AI-human boundary object
  - (small-data + big-models) > (big-data + small-models)

- **Prototyping, LLM configuration, and UX are key for responsible AI applications**
  - Biases are configured; you have to choose (& measure)
  - LLMs can "explain themselves" in natural language; this is often helpful, despite hot questions of fidelity
  - Rapidly evolving landscape of "vibe" coding

- **Jagged Frontier**
  - LLMs are amazingly smart AND amazingly dumb
  - Knowledge hugely changes what you can do with them

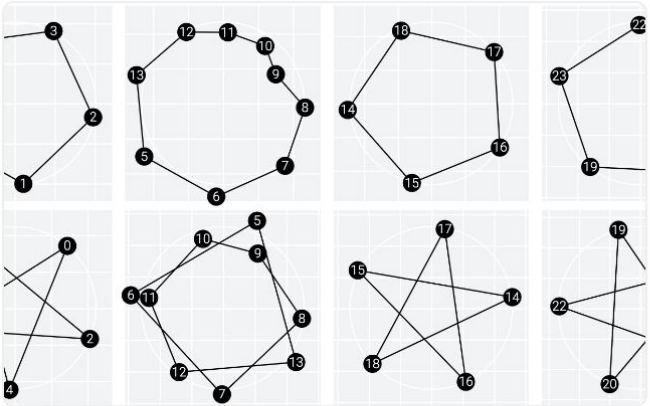


More...

https://arxiv.org/pdf/2303.12712.pdf

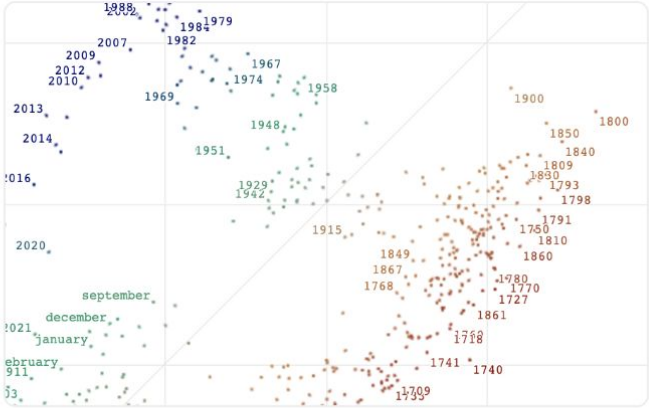
AI Explorables

Big ideas in machine learning, simply explained



Do Machine Learning Models Memorize or Generalize?

An introduction to grokking and mechanistic interpretability.



What Have Language Models Learned?

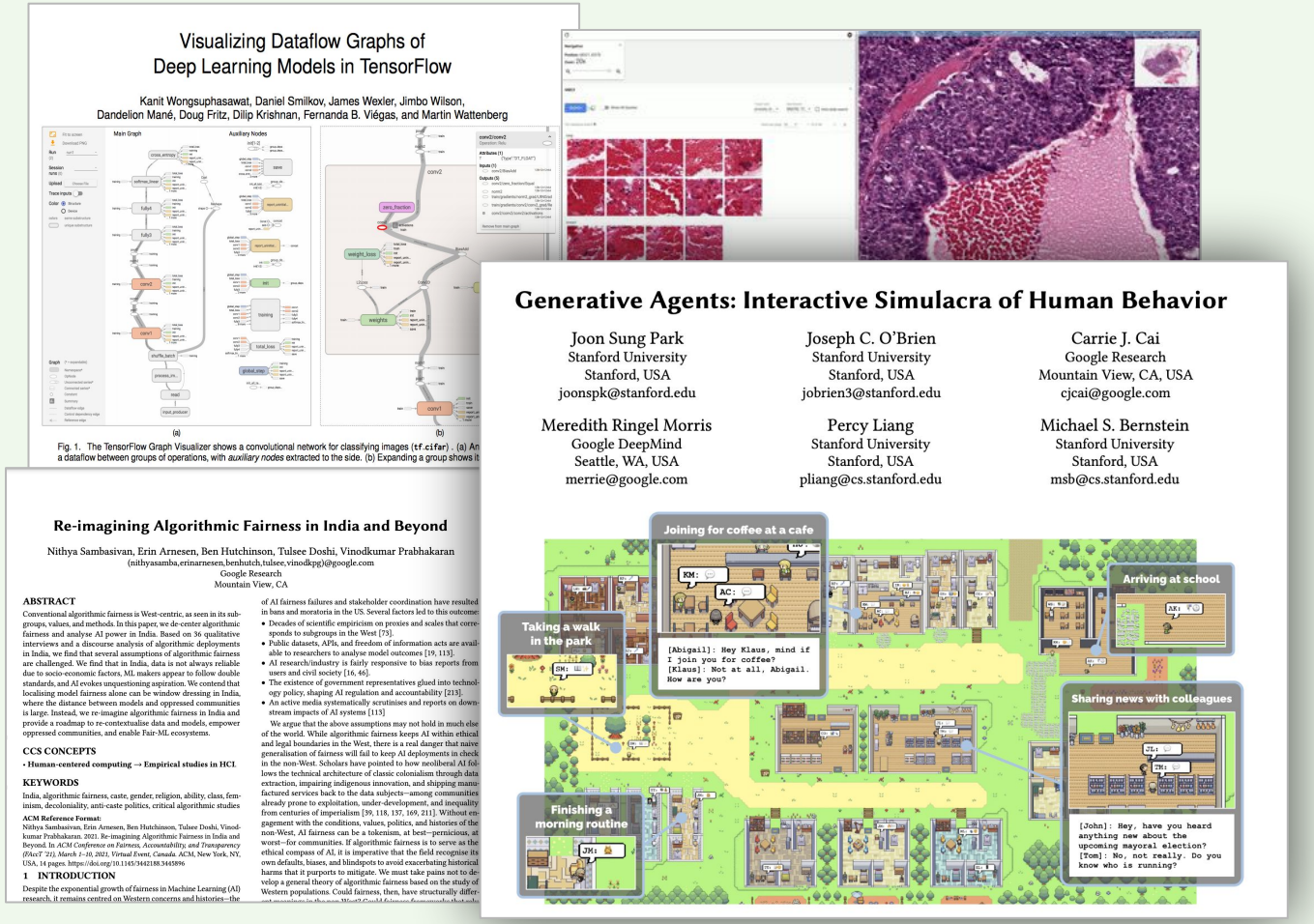
By asking language models to fill in the blank, we can probe their understanding of the world.

More at pair.withgoogle.com/explorables/

PAIR

Human-AI Interaction Research

with a recent focus being on large language models

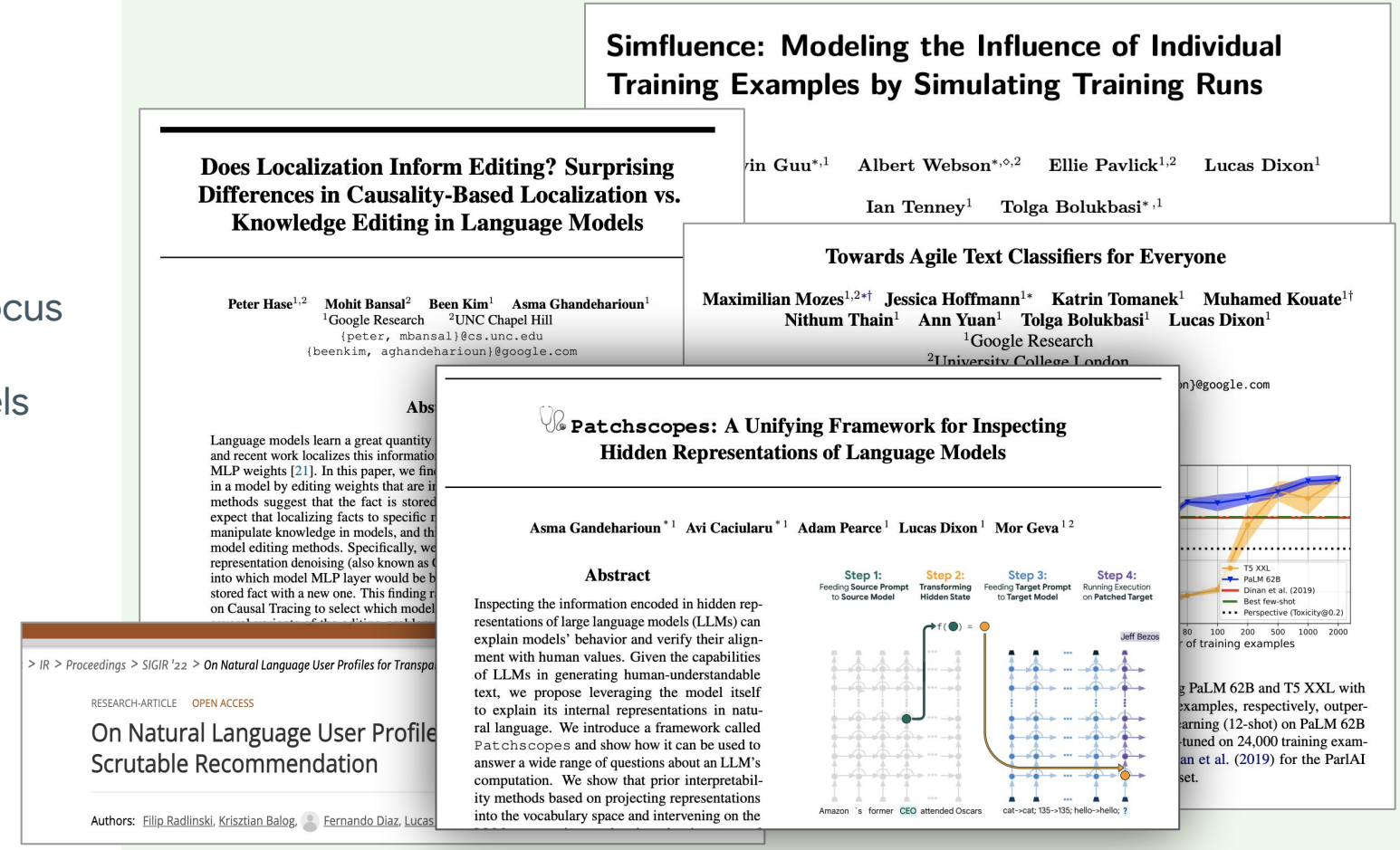


More at pair.withgoogle.com/research/

PAIR

ML Research

with a recent focus being on large language models



More at pair.withgoogle.com/research/

PAIR